While 루프와 For 루프에 대하여 배웁니다. 루프의 데이터 입력과 출력에 대하여 배웁니다. 타이밍 함수에 대하여 배웁니다.

CHAPTER 3. WHILE 루프와 FOR 루프

www.lvedu.kr

1. While 루프

루프 구조

While 루프

- While 루프는 루프 속의 코드를 반복 실행.
- 회색의 직사각형 틀 (화살표 모양)



www.lvedu.kr

구성 요소

- 루프 조건
 - 참인 경우 정지 ●



- 참인 경우 계속 →



- 반복 회수 터미널 🗓
 - 현재 루프의 반복 수를 나타냄
 - 0부터 시작하여 1씩 증가하는 l32 숫자형 값
 - 최대 값은 2,147,483,647
 - 최대 값을 초과하면, 실행이 계속되어도 반복횟수 터미널은 2,147,483,647로 유지됨

루프 실행 속도 조절

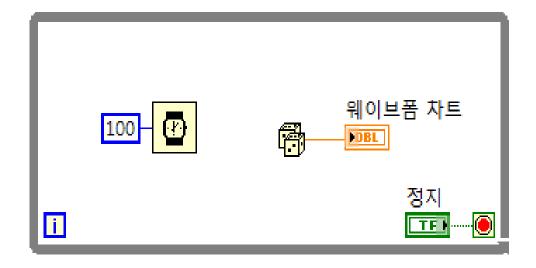
- 타이밍 함수 사용
 - 다음 ms 배수까지 기다림
 - 기다림 (ms)



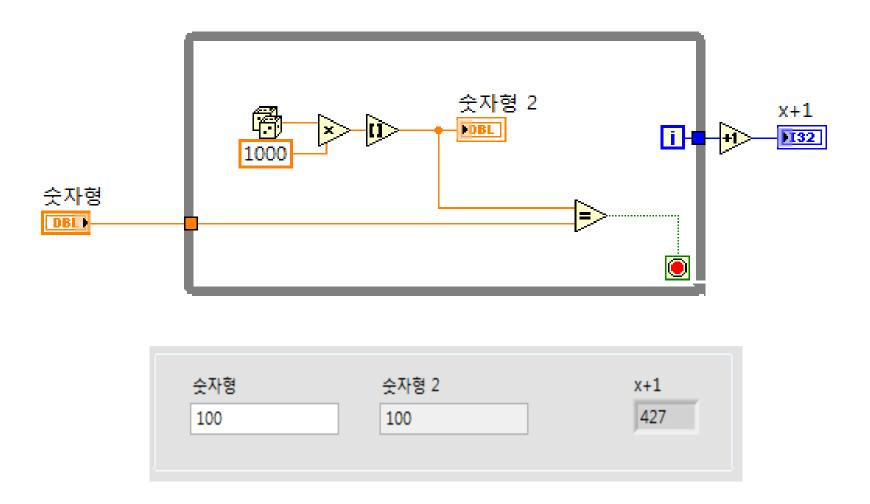




실습 3-1: 연속적인 난수 발생



실습 3-2: 난수 맞추기



2. For 루프

루프 구조

For 루프

- 반복 횟수 N만큼 루프 속의 코드를 반복 실행
- 메모지를 모아 놓은 모양



For 루프와 While 루프 비교

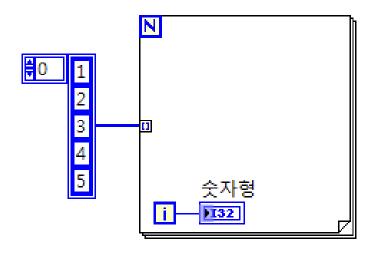
• For 루프는 몇 회 실행할 지, 미리 설정하고 루프 를 실행

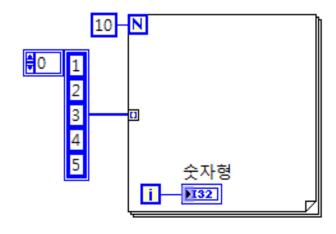
• While 루프는 몇 회 실행해야 될 지를 지정하지 않음 (정지 조건이 되면, 루프가 멈춘다.)

• For 루프는 N에 0을 입력하였을 때, 한번도 실행하지 않음. 그러나 While 루프는 반드시 1회는 실행해야 됨.

루프 카운트 입력

- 1. 루프 카운트에 I32 숫자형 값
- 2. 배열 입력을 통한 자동 인덱싱



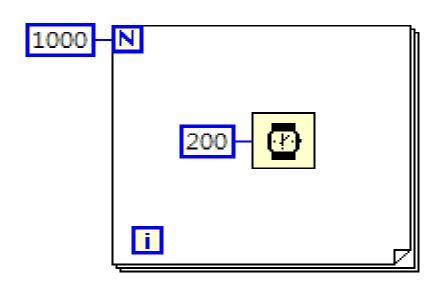


www.lvedu.kr

For 루프 타이밍 함수

- For 루프의 실행 속도를 조절함.
 - 기다림 (ms)

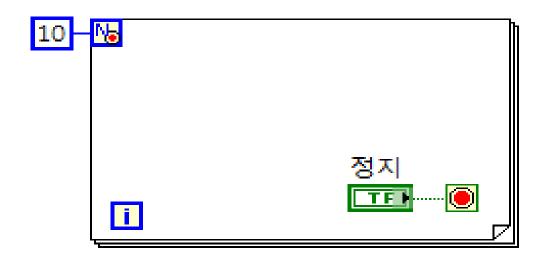




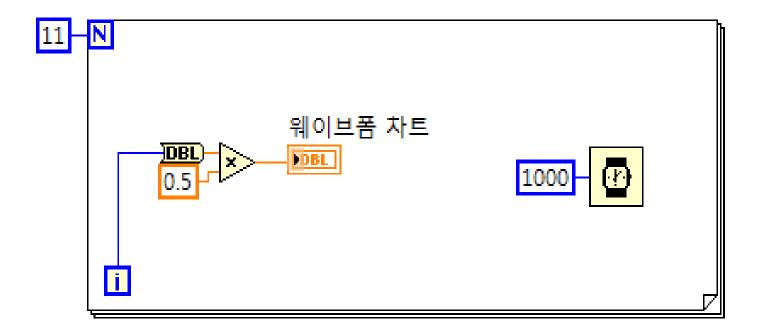
** 다음 ms 배수까지 기다림 함수는 While 루프에서만 사용합니다.

For 루프의 정지 조건

- For 루프 조건 터미널
 - 단축메뉴에서 [조건 터미널]을 선택하여, 조건 터미널을 For 루프에 추가할 수 있다.



실습 3-3: 계단식 증가 전압 만들기

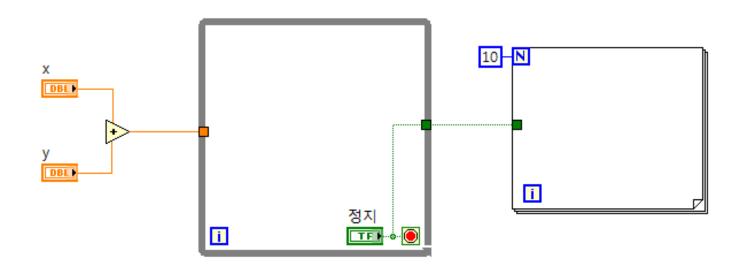


3. 루프 입력과 출력

루프 구조

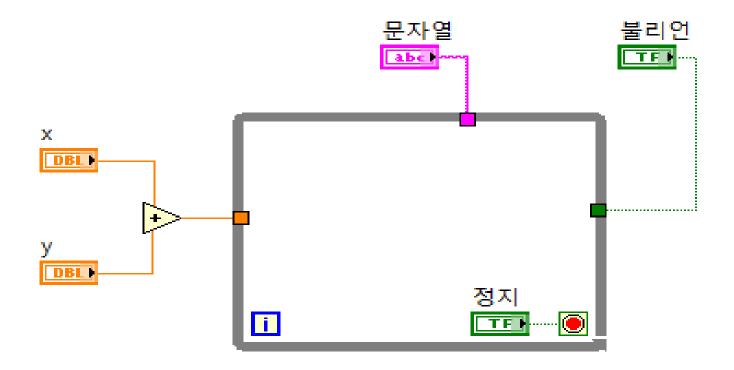
루프의 실행 조건

- LabVIEW의 코드는 와이어의 연결 순서에 따라서, 실행 순서가 결정된다.
 - 와이어 입력이 있는 함수나 노드, 루프는 해당 와이어로 데이터가 모두 입력되어야만 실행할 수 있다.



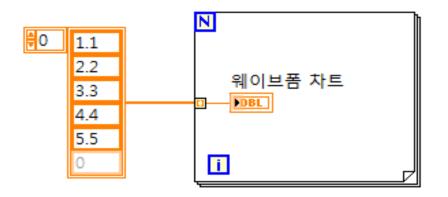
While 루프의 데이터 입력과 출력

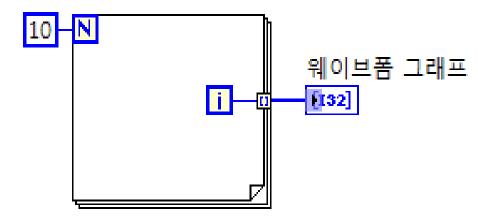
• 와이어가 연결된 위치와는 관계없이 컨트롤에서 루프로 데이터가 전달되는 흐름에 따라서 입력 여부가 결정된다.



For 루프의 자동 인덱싱

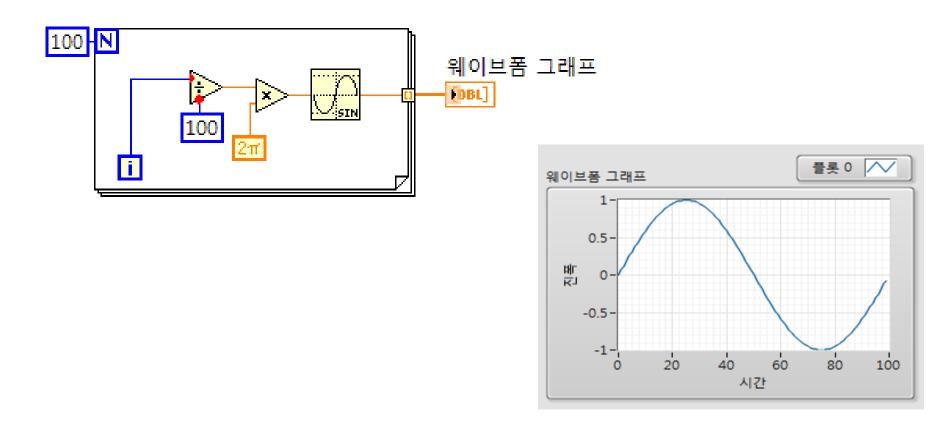
- 배열의 입력 > 자동 인덱싱으로 한 차원 낮춤
- 배열의 출력 > 자동 인덱싱으로 한 차원 높임.





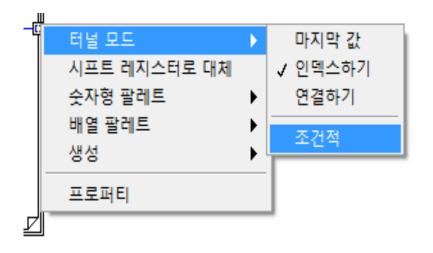
출력 자동 인덱싱

• 배열 만들기

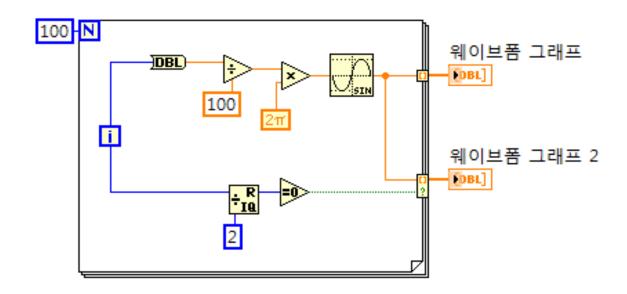


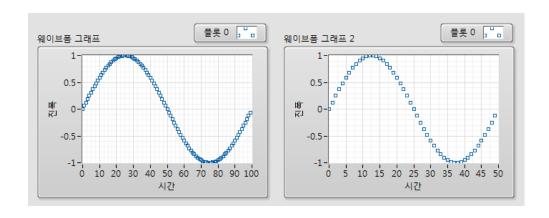
조건적 자동 인덱싱

• [단축메뉴] 터널 모드 > 조건적



실습 3-4: 조건적 자동 인덱싱





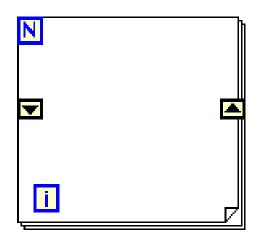
4. 시프트레지스터

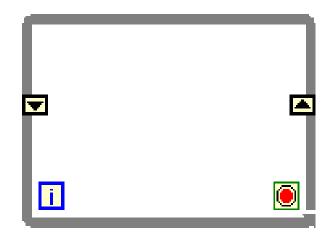
시프트레지스터

www.lvedu.kr

시프트레지스터 추가

- 루프 실행에서 다음 루프 실행으로 데이터 전달
- [단축메뉴] 시프트레지스터 추가

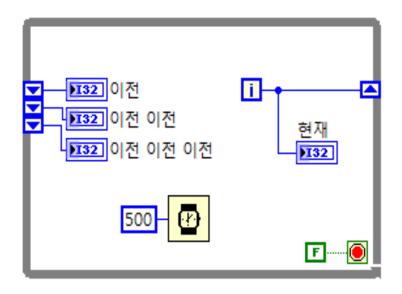


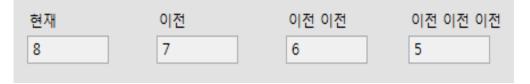


www.lvedu.kr

원소 추가

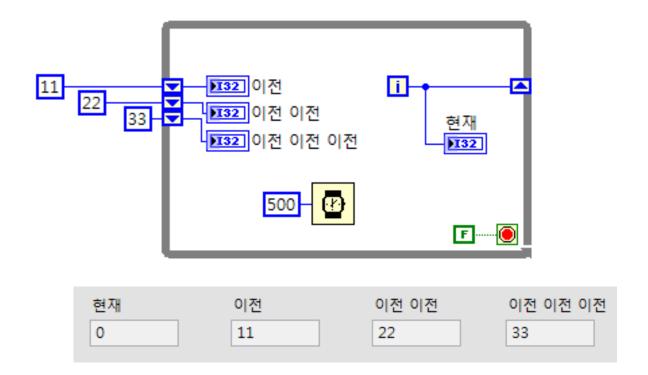
- 아래로 당겨서 늘이기
- [단축메뉴] 원소 추가





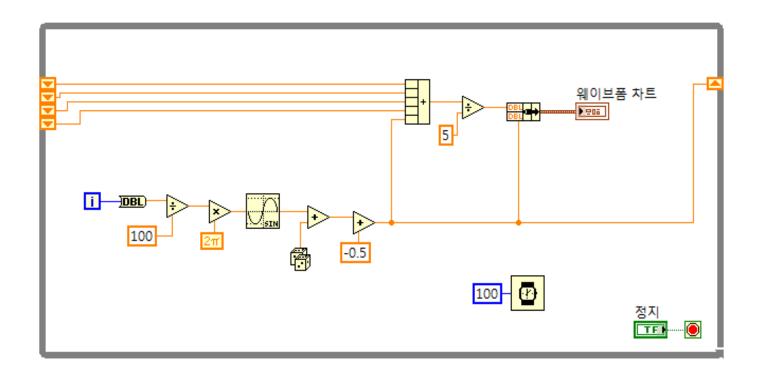
초기화

- 반드시 초기화해야 되는 것은 아닙니다.
- 초기화할 경우에는 모든 원소를 초기화 해줘야 됩니다.



실습 3-5. Running Average

• 이동 평균



5. 타이밍 함수

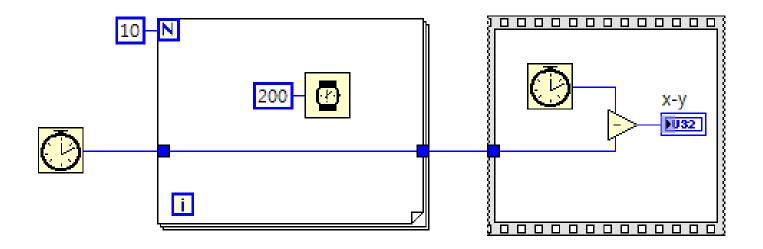
루프 속도 조절 시간 지연 시간 측정

www.lvedu.kr

틱 카운트(ms)

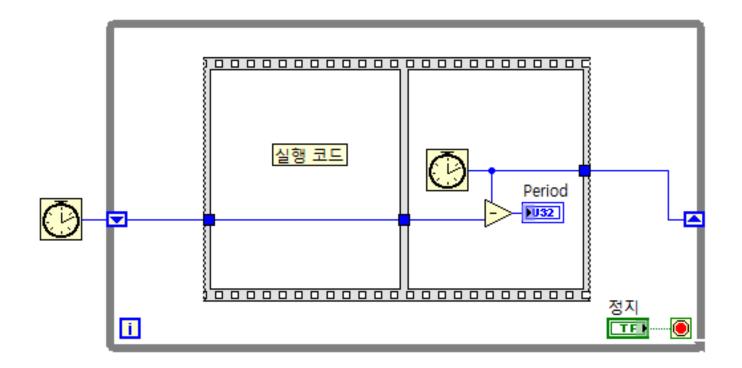


- msec 단위
- 0부터 2,147,483,647까지 카운트
- 터미널 카운트 2,147,483,647를 초과하면 다시 0부터 카운트
- 항상 틱 카운트 2개를 사용



루프 실행 속도 측정

• 시프트 레지스터를 이용하여 측정



www.lvedu.kr

높은 분해능의 상대 시간(초)

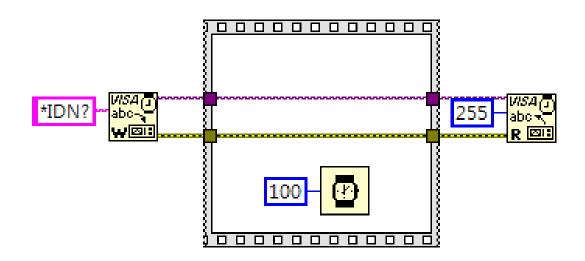


- LabVIEW 2014 버전에 추가됨
- 높은 분해능의 틱 카운트 함수

기다림 (ms)



- ms 단위의 실행 Delay를 주는 함수
- For 루프나 While 루프의 실행 속도를 조절하기 위한 용도 로 많이 사용
- 시퀀스 구조와 함께 사용하여 단계별로 Time Delay를 주 는 용도로도 많이 사용



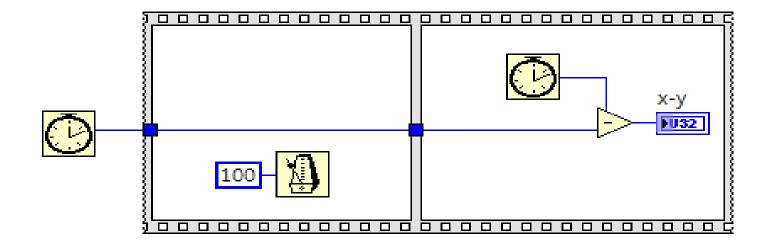
다음 ms 배수까지 기다림



- 메트로놈 기능을 구현
- While 루프에서만 사용
- 루프 속 코드의 실행 시간까지 고려하여, 전체 루프 속도 를 지정된 값으로 맞춤
- 루프 속 코드의 실행 시간이 타이머 값보다 길어질 경우에 는 다음 배수로 설정이 바뀜
- 다음 ms 배수까지 기다림 함수의 첫 번째 호출에서는 설 정 값보다 짧게 기다림

다음 ms 배수까지 기다림 (잘못)

- 한번 실행하는 경우에는 기다림 기능이 없음
- 타이밍 함수로 작동하지 않음

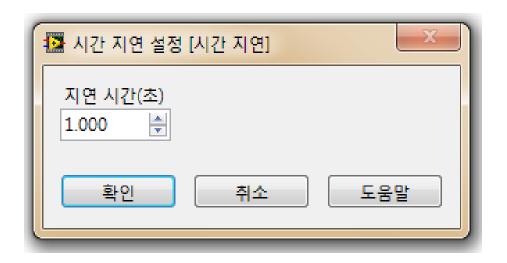


x-y 65

시간 지연



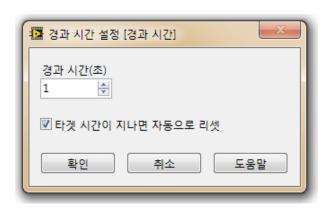
- 초 단위
- 팝업 설정 창이 지원되는 익스프레스 함수

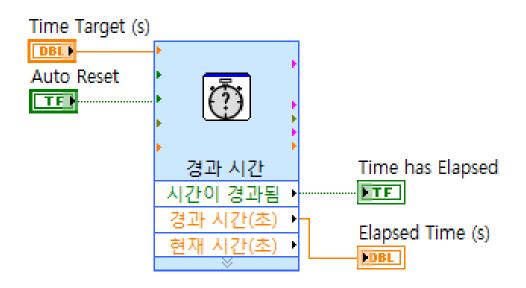


경과 시간



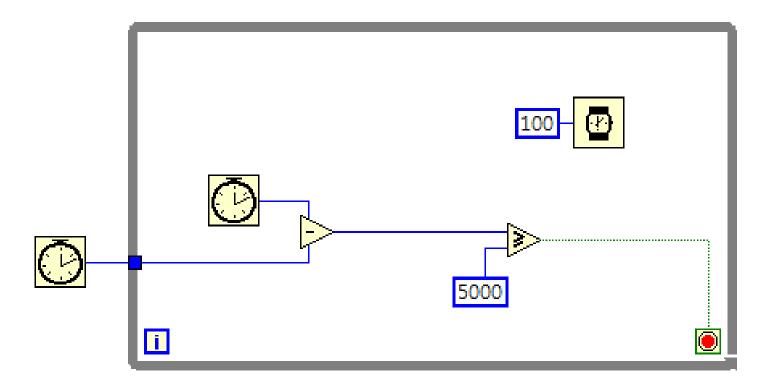
- 지정한 시작 시간부터 경과된 시간
 - 시간이 경과됨
 - 경과 시간
 - 현재 시간 (타임 스탬프)





시간 경과 후 루프 정지

• 틱 카운터(ms) 함수 이용



날짜/시간 문자열로 포맷

날짜/시간을 문자열로 얻기 날짜/시간 문자열로 포맷

www.lvedu.kr

날짜/시간을 문자열로 얻기



- 고정된 문자열 포맷입니다.
- 날짜 문자열은 YYYY-MM-DD 형식
- 시간 문자열은 오전/오후 HH:MM:SS 형식 ('초 보이기?'에 True를 입력해야 됩니다.)

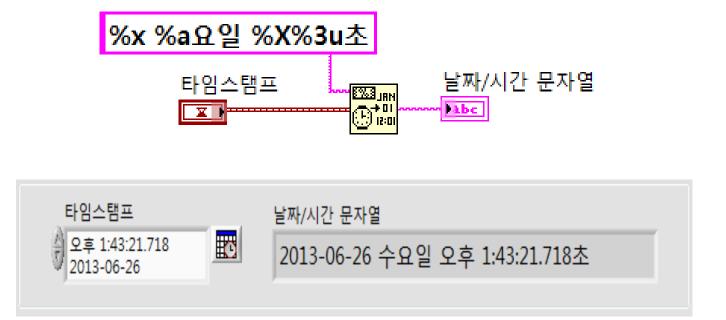




날짜/시간 문자열로 포맷



날짜/시간 문자열로 포맷 함수는 시간 포맷 코드를 사용하여 타임스탬프 또는 숫자 값을 지정하는 시간 포맷으로 변환하다.



포맷	설명		예제
%c	날짜/시간 정보	YYYY-MM-DD 오전/오후 시:분:초	2013-06-25 오후 9:19:03
%X	시간 정보	오전/오후 시:분:초	오후 9:19:03
%I	시	12시간	09
%Н	시	24시간	13
%M	분	분	19
%S	초	초	03
%<숫자>u		<숫자>정밀도를 가진 소수점 초	.444
%p	오전/오후	오전/오후	오후
%x	날짜	YYYY-MM-DD	2013-06-25
%y	축약된 연도	YY	13
%Y	연도	YYYY	2013
%m	달	MM	06
%b	축약된 달	М	6
%d	날짜	DD	25
%a	요일	요일	화

타임스탬프 상수

타임스탬프의 기본값은 세계시를 기준으로 1904년 1월 1일 금요일 오전 12시부터 0초가 지난 시간입니다[01-01-1904 00:00:00]. 시스템의 절대시간 값을 읽어온다.

