

현장 기술자를 위한

# 소형 PC에 의한 시퀀스 제어



현·장  
기·술

10

출판홍보과

## 제8장 래더 다이어그램으로부터의 프로그래밍

### 8.2 기본요소의 프로그램

입출력 및 일시기억 개소의 번지 할당이나 프로그램을 실행하여 시퀀스 동작을 확인하는 방법은 이미 제4장 「논리연산과 프로그래밍」에서 기술한 바 있다. 따라서 여기서는 래더 다이어그램을 구성하는 기본요소의 프로그래밍 방법에 대해서만 취급하기로 한다.

#### (1) 직렬접속회로

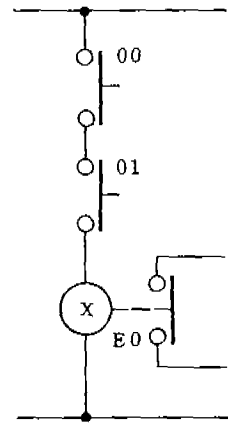
입력 00번지 접점과 01번지 접점이 그림 8.10과 같이 직렬로 접속되고 E0번지 접점에 출력되는 그림 8.11의 회로에 대해서 생각해 본다. 이와 같은 회로를 직렬접속회로라고 하는데, 논리회로에서는 이것을 AND회로라고도 하였다.

표 8.13의 프로그램 리스트로 프로그래밍을 생각해 보기로 하자.

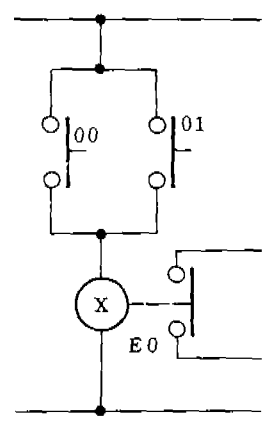
- 00 스텝 LD 00  
어드레스에서 지정하는 번지(00)의 내용을 판독한다.
- 01 스텝 AND 01  
어드레스에서 지정하는 번지(01)의 내용과 먼저 판독한 내용이 직렬접속, 즉 AND로 되어 있으므로 이와 같이

<표 8.13> 프로그램 리스트

스텝 No.	명령		비고
	OP	ADD	
00	LD	00	
01	AND	01	
02	OUT	E0	



<그림 8.10> 직렬접속회로



<그림 8.11> 병렬접속회로

<표 8.14> 프로그램 리스트

스텝 No.	명 령		비 고
	OP	ADD	
00	LD	00	
01	OR	01	
02	OUT	E0	

된다.

02 스텝 OUT E0

출력번지(E0)에 위의 결과를 출력한다.

(2) 병렬접속회로

다음에 입력 00번지 접점과 01번지 접점이 그림 8.11과 같이 병렬로 접속되어 E0번지 접점에 출력되는 그림 8.11의 회로에 대해서 생각해 본다. 이와 같은 회로를 병렬접속회로라고 하는데, 논리회로에서는 이것을 OR회로라고도 불렀다.

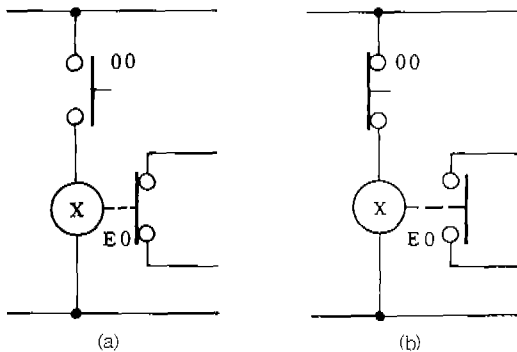
표 8.14의 프로그램 리스트로 프로그래밍을 생각해 보자. 이 프로그램에서 앞의 (1) 직렬접속회로의 프로그램과 다른 것은 다음의 01 스텝만이다.

01 스텝 OR 01

01번지의 내용과 먼저 관독한 00번지의 내용이 병렬접속, 즉 OR로 되어 있으므로 이와 같이 된다.

(3) b접점회로

논리회로에서는 이것을 NOT회로라고 불렀지만 래더 다이어그램에서는 그림 8.12, 그림 8.13의 회로를 b접점회로라고 한다. 이 회로는 입력을 푸시버튼 스위치의 접점으로 부여하는 경우(그림 8.12



<그림 8.12> b접점회로(1)

<표 8.15> 프로그램 리스트

(a)			(b)		
스텝 No.	명 령		스텝 No.	명 령	
	OP	ADD		OP	ADD
00	LD	00	00	LD	00
01	OUT·NOT	E0	01	OUT	E0

<표 8.16> 프로그램 리스트

(a)			(b)		
스텝 No.	명 령		스텝 No.	명 령	
	OP	ADD		OP	ADD
00	LD	10	00	LD·NOT	10
01	OUT·NOT	E0	01	OUT	E0

의 경우)와 릴레이 접점(그림 8.13의 경우)이 있는데, 양자는 프로그램이 다르므로 주의를 요한다.

그림 8.12(a)는 입력에 푸시버튼 스위치의 a접점을, 출력측에 릴레이의 b접점을 사용한 예이다. 이 회로를 프로그램한 것이 표 8.15(a)이다. 01 스텝에 주의하기 바란다.

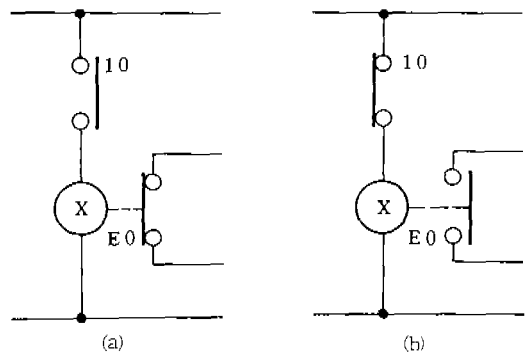
01 스텝 OUT·NOT E0

이것에 대해서 그림 8.12(b)는 입력에 푸시버튼 스위치의 b접점을 사용하고 출력측을 릴레이의 a접점을 사용한 예이다. 프로그램 예가 표 8.15(b)에 표시되어 있다. 00 스텝에 주의하기 바란다.

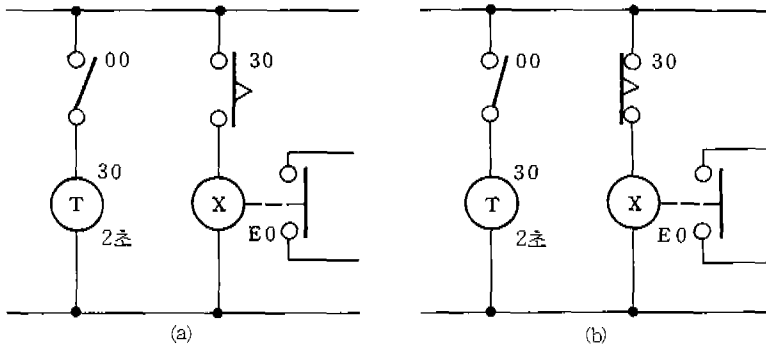
00 스텝 LD 00

여기서 입력은 b접점으로 부여하고 있기 때문에 LD·NOT로 하고 싶지만 그렇지 않다. 이유는 제4장의 NOT 회로 설명을 참조하기 바란다.

다음에 그림 8.13(a), (b)는 입력이 릴레이 접점



<그림 8.13> b접점회로(2)



〈그림 8.14〉 한시접점회로

으로 부여되는 경우이다. 입력접점 10번지의 릴레이 코일은 이 회로의 전단에 있는 것으로 한다. 접점 10번지는 릴레이 코일의 여자상태에 의해 작동하는 것이다. 표 8.16(a), (b)에 각각의 프로그램이 표시되어 있다. 앞의 예(그림 8.12(a), (b), 표 8.15(a), (b))와 비교해 보자.

(4) 한시접점회로

한시동작 순시 복귀의 a접점회로를 그림 8.14(a)

에, 동 b접점회로를 그림 8.14(b)에 나타낸다. 그리고 각각의 프로그램을 표 8.17(a), (b)에 들었다. 번지 및 타이머 시간의 설정은 앞의 8.1 「타이머의 취급」에서 기술한 바와 같다. 그리고 기타 프로그램 만들기에 관해서도 문제가 없다.

여기서는 한시동작의 a접점회로 및 b접점회로를 나타내는 프로그램이 소정의 동작을 하는가의 여부를 확인하고 한시동작회로의 프로그램을 만드는 방법을 확실히 하여 두자.

지금까지의 명령의 정리

명 령	OP	ADD	기 능
LOAD	LD	입력 00-0F번지	ADD로 지정하는 내용의 관독
	LD · NOT	출력 E0-EF번지	ADD로 지정하는 내용의 부정 관독
		일시기역 10-3F번지	
STORE	STO	일시기역 10-3F번지	일시기역
OUT	OUT	출력 E0-EF번지	ADD로 지정하는 출력단자에의 출력
	OUT · NOT		ADD로 지정하는 출력단자에 부정하여 출력
AND	AND	입력 00-0F번지	논리적
	AND · NOT	출력 E0-EF번지	논리적 부정
		일시기역 10-3F번지	
OR	OR	입력 00-0F번지	논리적
		출력 E0-EF번지	
	OR · NOT	일시기역 10-3F번지	논리화 부정
NOT	INV	00	부정
TIMER	TIM	시간설정* 1~F	한시동작지정, 시간설정
		일시기역 (3)0-(3)F번지	
NOP	NOP	00	아무 것도 실행하지 않고 다음 스텝으로 진행

\* 타이머의 지정, 시간 설정

ADD의 처음 숫자는 시간설정(1~F)

설정시간 T(초)는 다음 식으로 계산:  $T = \text{설정값}(1 \sim F) \times 0.5$

ADD의 다음 숫자는 타이머의 일시기역 번지의 지정(30번지의 1자리째의 0~F만 기록한다)

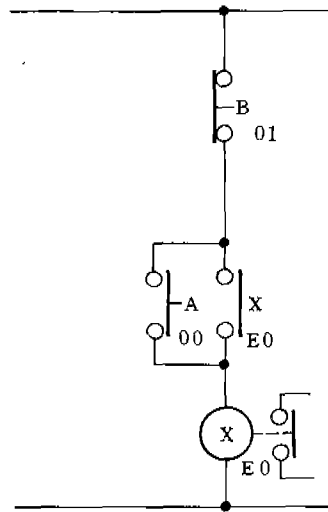
〈표 8.17〉 프로그램 리스트

(a)				(b)			
스텝 No.	명령		비고	스텝 No.	명령		비고
	OP	ADD			OP	ADD	
00	LD	00	2초	00	LD	00	2초
01	TIM	40		01	TIM	40	
02	LD	30		02	LD·NOT	30	
03	OUT	E0		03	OUT	E0	

## 제9장 래더 다이어그램 기본회로의 프로그래밍과 실행

제5장에서는 기본 시퀀스 제어회로의 프로그래밍과 그 실행에 대해서 기술하였다. 제5장에서의 취급은 유점점회로도 동시에 표시하고 있지만 어디까지나 로직 다이어그램에서 프로그램을 작성하는 것에 주안이 두어지고 있다.

여기서 취급하는 래더 다이어그램으로 표현한 기본회로의 프로그래밍에 관해서도 입출력 및 일시기억 번지의 할당방법이나 프로그램을 실행시킨 동작 확인과 같은 것은 제5장과 동일하다. 이 장의 주요 목적은 제8장에서 기술한 래더 다이어그램에서의 프로그래밍의 특징을 이해하고 래더 다이어그램에서 직접 프로그램을 작성하기 위해 연구되고 있는 사항을 이곳의 기본회로 프로그래밍에 활용해 보는 것에 있다.



〈그림 9.1〉 리셋 우선 자기유지회로

### 9.1 자기유지회로

그림 9.1에 리셋 우선 자기유지회로를 나타낸다. 유지점점 X는 출력 X와 연동하게 되어 있다. 여기에는 제8장의 8.1 「출력 OUT 데이터를 점점 데이터로서 사용」에서 기술한 사항이 활용된다. 표 9.1의 프로그램 리스트를 보기로 하자.

00 스텝 LD 00

병렬회로측부터 프로그램을 만들어 나간다. 전기회로에 있어서의 저항의 직병렬회로의 계산을 병렬회로부터 직렬회로로 진행하는 것과 동일한 요령이다.

〈표 9.1〉 프로그램 리스트

스텝 No.	명령		비고
	OP	ADD	
00	LD	00	
01	OR	E0	
02	AND	01	
03	OUT	E0	

01 스텝 OR E0

점점 X(E0번지)가 점점 A와 병렬 접속, 즉 OR로 되어 있다.

출력 OUT의 데이터(E0번지의 내용)를 이 점점 X의 데이터로서 사용하고 있으므로 ADD부가 E0가 된다.



의 연산처리 결과를 10번지에 격납해 둔다.

- 04 스텝 TIM 40
- 05 스텝 LD·NOT 30

프로그램의 흐름을 일단 중단하고 여기서 타이머의 시간 설정을 하고 한시동작의 b접점(30번지)을 판독한다.

- 06 스텝 AND 10

여기서 판독한 타이머 접점(30번지)과 03 스텝까지의 연산결과(10번지에 격납되어 있다)가 직렬로 연결되어 있으므로(AND) 10번지와 AND를 취하게 된다.

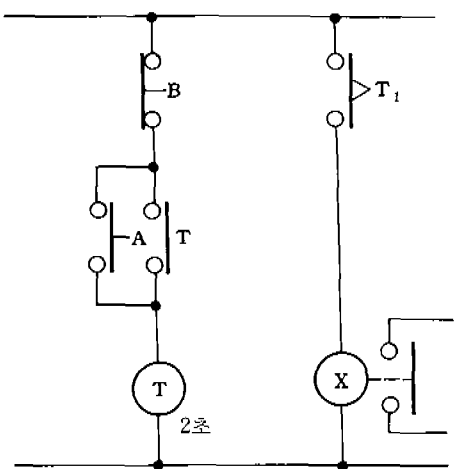
- 07 스텝 OUT E0

E0번지에 출력함과 동시에 이 번지의 내용은 유지접점의 데이터로서도 사용되고 있다.

**연습문제 9.2**

그림 9.4에 지연동작회로가 있다. 이 회로와 동일한 동작을 하는 프로그램을 작성하시오

다만 이 회로의 타이머는 순시동작접점 T<sub>1</sub>와 한시동작접점 T와의 양쪽 접점을 가지는 것이 사용되고 있다. 그래서 이것을 PC의 프로그램으로 하는 때는 순시동작 접점은 보조 릴레이 코일(X<sub>0</sub>)로

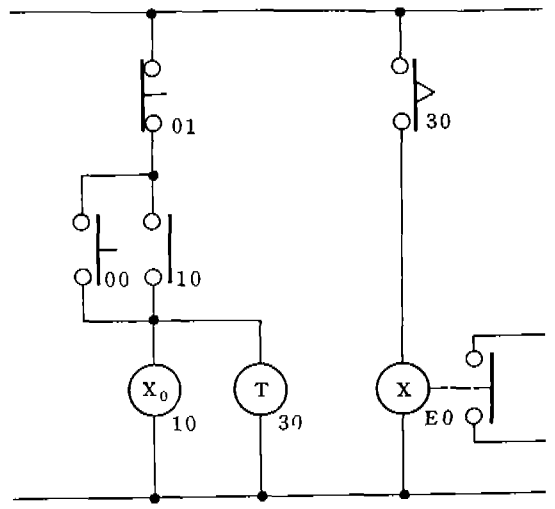


<그림 9.4> 지연동작회로

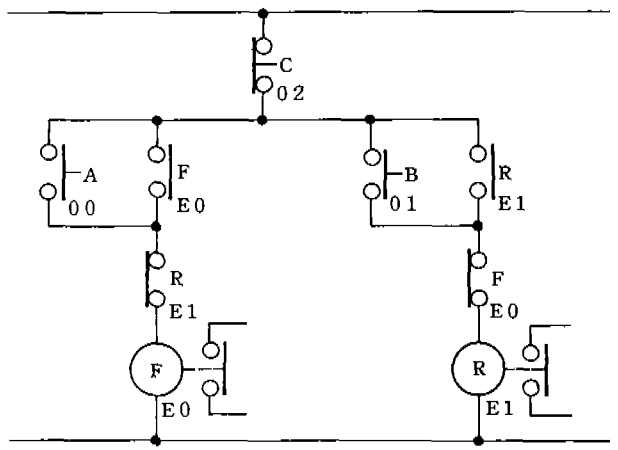
동작시키고 한시동작접점만을 타이머 코일 ①로 동작시키는 그림 9.5의 회로에 일단 바꾸어 그리고 나서 프로그램하면 알기 쉽다.

**9.3 인터록회로**

그림 9.6에 인터록회로를 나타낸다. 여기서는 연산처리 대기가 생기지 않도록 프로그램을 만드는 것과 복합명령 사용방법 및 출력 OUT의 데이터를 접점 데이터로서 사용하는 것의 세가지 점을 염두에 두고 표 9.3의 프로그램 리스트를 보기로 하자.



<그림 9.5> 지연동작회로(그림 7.5)를 바꾸어 그린 것



<그림 9.6> 인터록회로

<표 9.3> 프로그램 리스트

스텝 No.	명령		비고
	OP	ADD	
00	LD	00	
01	OR	E0	
02	AND	02	
03	AND·NOT	E1	
04	OUT	E0	
05	LD	01	
06	OR	E1	
07	AND	02	
08	AND·NOT	E0	
09	OUT	E1	

00 스텝 LD 00

점점 00 및 E0의 병렬회로에서 점점 02 및 E1의 직렬회로로 프로그램을 만들고 연산처리 대기가 생기지 않도록 한다.

01 스텝 OR E0

출력 OUT의 메이터(E0번지의 내용)를 유지용 점점번지로서 사용하고 있다.

02 스텝 AND 02

03 스텝 AND·NOT E1

02, E1 점점이 직렬접속으로 AND이다. E1 점점은 릴레이 코일 ⑧에 의해 동작하는 b점점이므로 복합명령을 사용하여 AND·NOT로 한다.

04 스텝 OUT E0

F(E0번지)에 출력하여 이것으로 좌

측회로의 프로그램이 완료된다.

우측회로에 대해서는 좌측회로와 대칭으로 동일한 요령으로 프로그램된다.

### 9.4 한시동작의 이행회로

그림 9.7에 한시동작의 이행회로를 나타낸다. 이 회로는 제어회로로서 일반적으로 표현하기 위해 한시동작의 이행 등으로 특수한 표현으로 하고 있지만 전술한 3상 유도전동기의 자동 Y-Δ 기동에 응용되는 것이었다.

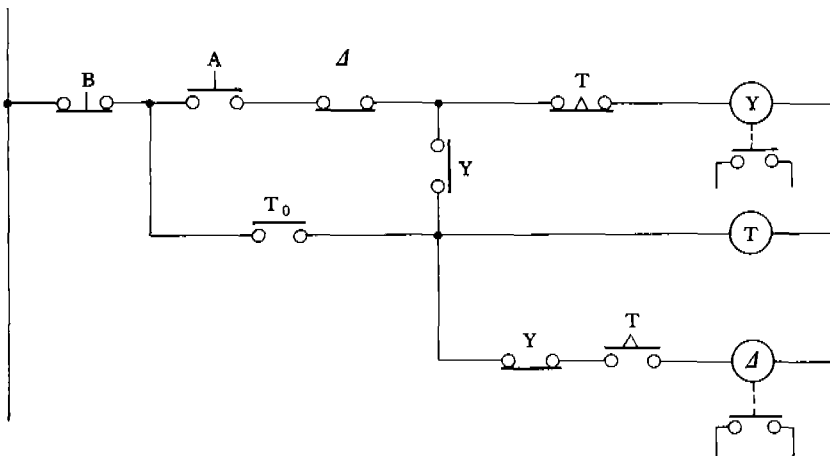
여기서의 프로그래밍에서 유의할 점은 다음 세 가지이다.

첫째는 회로를 프로그램하기 쉽도록 출력 ⑤ 코일, ④ 코일 각각을 중심으로 한 회로로 바꾸어 그리는 것이다(그림 9.8).

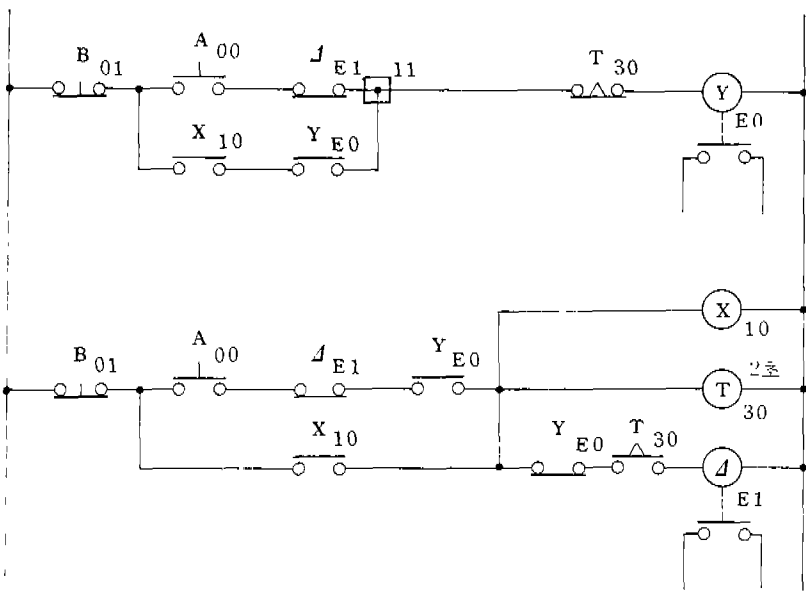
두번째는 연산처리 대기를 위한 일시기억 번지 11을 설정하는 것이다(그림 9.8 ⑥ 코일측의 회로 참조).

세번째는 (연습문제 9.2)에서 기술한 바와 같이 타이머 취급에 대해서 고려하는 것이다. 그림 9.7의 회로의 타이머는 순시동작 점점 T<sub>0</sub>와 순시동작 점점 T 양 쪽 점점을 가지는 것이 사용되고 있으므로 이것을 바꾸어 써서 그림 9.8과 같이 한다.

즉, 그림 9.8의 ④ 코일측 회로에 나타내듯이 순시동작 점점은 보조 코일 ⑩로 동작시키고 한시동작 점점만을 타이머 코일 ⑪로 동작시키게 하는 것이다. ⑩는 일견 출력과 같지만 그렇지 않고 보조 릴레이의 역할을 수행하고 있는 것이다. 그



<그림 9.7> 한시동작의 이행회로



<그림 9.8> 한시동작의 이행회로(그림 7.7)을 바꾸어 그린 것

<표 9.4> 프로그램 리스트

스텝 No.	명 령		비고
	OP	ADD	
00	LD	01	
00	LD	00	
01	AND·NOT	E1	
02	STO	11	
03	LD	10	
04	AND	E0	
05	OR	11	
06	AND	01	
07	AND·NOT	30	
08	OUT	E0	
09	LD	00	
0A	AND·NOT	E1	
0B	AND	E0	
0C	OR	10	
0D	AND	01	
0E	STO	10	
0F	TIM	40	2초
10	AND·NOT	E0	
11	AND	30	
12	OUT	E1	

때문에 이 코일의 상태는 일시기억(10번지)으로 설정해 둘 필요가 있었다. 이 ㊸코일 상태의 데이터(10번지의 내용)가 점점 ㊸의 내용으로서 사용되고 있기 때문이다.

이상에 유의하여 표 9.4의 프로그램 리스트를 보기로 하자.

02 스텝 STO 11  
두번째의 유의점에서 기술한 연산 처리 대기를 위해 일시기억 11번지에 연산결과를 격납해 둔다.

08 스텝 OUT E0  
이것으로 ㊸코일을 중심으로 한 상반 회로의 프로그램이 완성된다.

0E 스텝 STO 10  
세번째 유의점에서 기술한 보조 코일 ㊸의 상태를 일시기억 10번지에 격납한다.

0F 스텝 TIM 40  
타이머 코일 ㊸의 상태도 보조 코일 ㊸의 상태와 동일하므로 계속해서 이와 같이 처리한다.

**연습문제 해**

<문제 9.1의 해답> 프로그램 리스트

스텝 No.	명 령		비 고
	OP	ADD	
00	LD	01	
01	AND	E0	
02	OR	00	
03	OUT	E0	

<문제 9.2의 해답> 프로그램 리스트

스텝 No.	명 령		비 고
	OP	ADD	
00	LD	00	
01	OR	10	
02	AND	01	
03	STO	10	
04	TIM	40	
05	LD	30	
06	OUT	E0	

☆ ★ ☆

이상, 10회에 걸쳐 연재해 온 「현장 기술자를 위한 소형 PC에 의한 시퀀스 제어」를 끝내기로 한다. 이번 관계상 래더회로 프로그램 응용예를 소개하지 못하는 것을 유감으로 생각한다.