

공장자동화에 있어서 프로그램식 제어기의 역할

(3)

강 영 채

전 인천대학교 교수

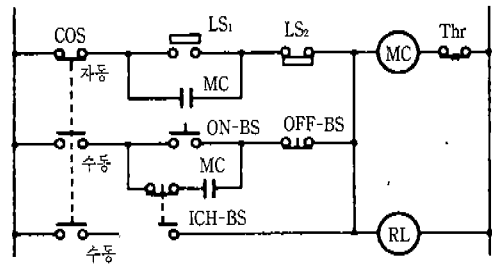
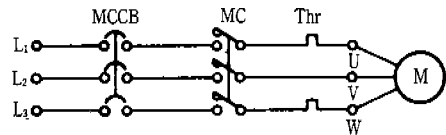
4. PC에 의한 3상 유도전동기의 운전

생산현장에서 사용되고 있는 모터는 유도기, 직류기 및 서보모터 등과 같은 여러 가지가 있으나 본고에서는 프로그램식 제어기(PC)에 의한 3상 유도전동기의 始動, 運轉, 停止방법을 설명하고자 한다.

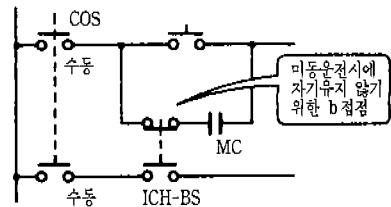
4.1 全電壓 및 微動運轉

전전압 시동은 소용량(2.2kW, 3마력 정도)의 전동기에 사용되는 시동법이며 전원전압을 모터의 단자(U.V.W)에 직접 인가해서 시동과 운전을 한다.

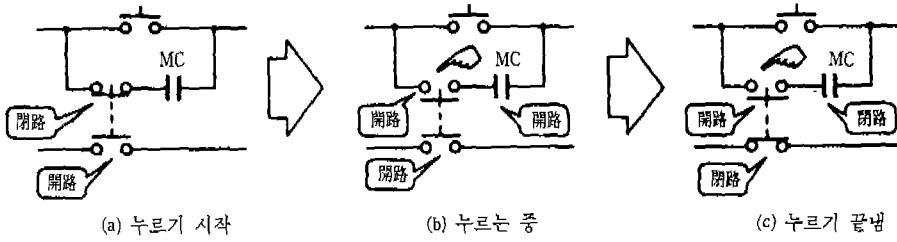
미동운전이란 미동버튼을 누르는 동안에만 모터가 동작하는 것으로서 인칭이라고도 한다. 자동-수동을 전환할 수 있는 릴레이 시퀀스회로의 예를 그림 1에 그리고 그림 2에 미동회로의 일부를 나타낸다.



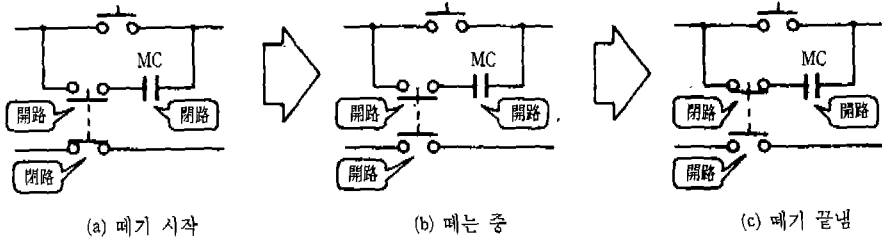
<그림 1> 전동기의 전전압 시동회로



<그림 2> 미동용 BS



<그림 3> 미동시작 지령



<그림 4> 미동해제 지령

여기서 미동용 버튼스위치 ICH-BS는 직접 PC에 적용되지 않으며 릴레이 시퀀스에서 버튼스위치를 눌렀을 때 b접점이 自己維持回路를開放한다.

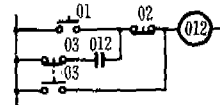
이러한 과정을 그림 3, 4에 나타내고 있는데 2개 그림의 (b)에서 a접점(메이크접점)이나 b접점(브레이크접점)은 개방하는 시간이 있으며 이 시간을 이용하여 자기유지를 하지 않도록 하고 있다.

이와 같은 릴레이 시퀀스회로를 그대로 PC 회로에 적용하면 그림 5와 같이 되며 이 회로에서 ICH-BS를 누른 뒤에 떼어도 MC는 그대로 자기유지를 한다.

한편 기계적인 동작을 하는 ICH-BS의 a접점과 b접점의 전환시간 지연을 PC로 바꾸어 놓으면 프로그램처리가 너무 빨라서 기대하는 동작을 얻지 못하게 된다.

그림 6은 데이터와 레지스터의 내용 및 프로그램의 연산결과를 나타낸다. 그림 6(d)와 같이 미동버튼이 OFF "0"으로 되어도 OUT12가 ON "1"

어드레스	명 령	데이터
00	LD	01
01	LD NOT	03
02	AND	012
03	OR LD	-
04	AND NOT	02
05	LD	03
06	OR LD	-
07	OUT	012
08	END	-



(a) 시퀀스

(b) 코딩

<그림 5> 미동 프로그램(에러)

로 되어 있다.

이와 같이 릴레이 시퀀스에서 접점의 시간지연을 이용하여 회로를 구성하는 경우에는 그대로 PC에 적용할 수 없는데 이때는 그림 7과 같이 타이머로서 a접점이나 b접점중 어느 한쪽 접점의 동작시간을 강제 지연시켜야 한다.

그림 1의 회로를 PC에 적용한 회로의 예를 그

명령	데이터	레지스터	
		R	S
LD	01	0	-
LD NOT	03	1	0
AND	012	0	0
OR LD	-	0	-
AND NOT	02	0	-
LD	03	0	0
OR LD	-	0	-
OUT	012	0	-

(a) 초기상태(012 : OFF)

데이터	레지스터	
	R	S
0	0	-
0	0	0
0	0	0
-	0	-
1	0	-
1	1	0
-	1	-
1	1	-

(b) 미동버튼 ON

데이터	레지스터	
	R	S
0	0	-
0	0	0
1	0	0
-	0	-
1	0	-
1	1	0
-	1	-
1	1	-

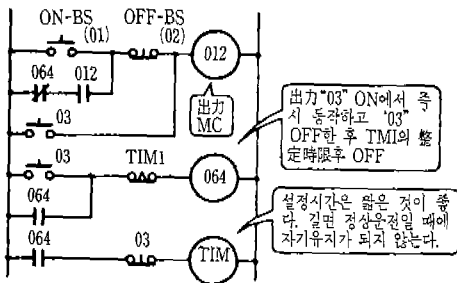
(c) ON 계속중

데이터	레지스터	
	R	S
0	0	-
1	1	0
1	1	0
-	1	-
1	1	-
0	0	1
-	1	-
1	1	-

(d) 미동버튼 OFF

(b) 이하의 데이터는 각각의 접점이나 ON-OFF를 1과 0으로 나타낸다.

<그림 6> 데이터와 레지스터의 내용

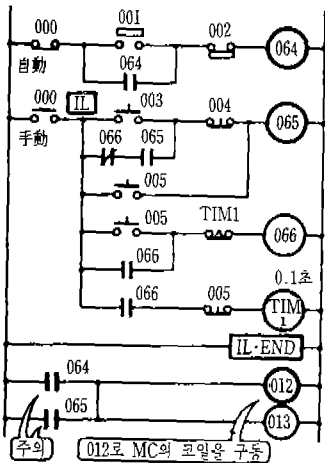


<그림 7> OFF 릴레이 타이머 사용

림 8에 나타내었다. 여기에는 IL/IL.END 명령이 수동-자동 전환회로 부분에 사용되고 있는데 이것은 블록 사이를 구별하는 명령으로서 마스터 컨트롤러라고도 한다.

그런데 자동 "064"와 수동 "065"의 내부보조 릴레이에 있는 2개의 접점을 병렬로 사용하여 최종출력 "012"의 MC를 구동한다.

그림 9는 수동-자동, 전전압-미동운전회로의 외부배선에 대한 한 가지 예로서 입력측은 모두 a 접점에 접속시켜야 한다. 그 첫째 이유는 시퀀스회로(그림 8)상에서 b 접점이 사용되고 있는



(a) 시퀀스

어드레스	명령	데이터
00	LD	001
01	OR	064
02	AND NOT	000
03	AND NOT	002
04	OUT	064
05	LD	000
06	IL	-
07	LD OUT	066
08	AND	065
09	OR	003
10	AND NOT	004
11	LD	005
12	OR LD	-

어드레스	명령	데이터
13	OUT	065
14	LD	005
15	OR	066
16	AND NOT TIM	1
17	OUT	066
18	AND NOT	005
19	TIM1	001
20	IL END	-
21	LD	064
22	OR	065
23	OUT	012
24	END	-

(b) 코딩

<그림 8> 미동 운전회로

기기로서 PC의 입력측에 접속하는 것이 일반적이거나 이와 같이 하지 않는 것이 오히려 좋은 경우가 있다(그림 12 참조).

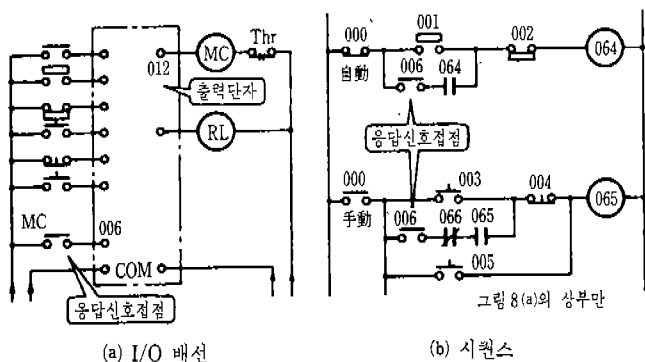
이유는 제어 그 자체가 PC의 내부에서 논리적으로 처리된 다음에 출력측에 있는 MC(Electromagnetic Contact)의 코일을 消磁하는 것이 아니고 직접 코일 MC를 소자하는 것이 신뢰성을 향상시키기 때문이다.

그러나 그림 10을 그대로 사용하면 문제가 발생하게 된다. 만일 모터의 과부하로 인하여 열동계전기(Thr)가 동작하고 코일 MC를 소자하였을 때에는 Thr의 접점이 개회로가 되는데 PC 내부의 연산처리는 여전히 출력단자(012)가 ON 상태가 된다.

한편 열동계전기(Thr)의 리세트버튼을 눌러서 복귀시킨 순간에 모터가 회전하면 위험하므로 이를 피하기 위해서는 외부회로만으로 구성되어 있는 열동계전기(Thr)를 개회로상태로 만들고 모터가 정지했을 때 PC 프로그램의 自己維持를 제거하여야 한다. 그림 13과 같이 응답신호 접점을 외부 입력측에 접속하고 프로그램에 도입할 필요가 있다.

세번째의 주의사항은 출력점수와 출력용량으로서 PC는 입출력 점수에 제한이 있기 때문에 출력 접점의 용량과도 관계가 있다.

만일 동일한 동작을 하는 출력기기이면 병렬접속에 의하여 사용하는 점수를 감소시키도록 하고



<그림 13> 응답신호

그림 14로서 접점용량을 설명하기로 한다. 즉 출력접점에 전자접촉기와 표시등을 접속하고 각각의 코일에 MC 여자전류가 30mA, SL의 통전전류가 80mA였다고 하자.

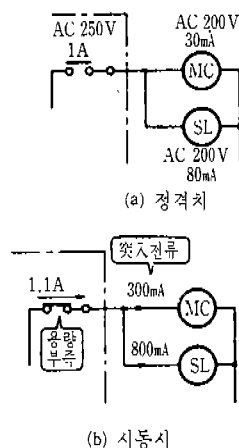
코일이나 램프는 시동시에 突入電流이라고 하는 정격전류의 8~10배 정도로 전류가 흐르는 성질이 있다.

이것은 출력접점이 닫혀진 순간에 접촉기의 코일은 300mA, 램프는 800mA 합계 1,100mA(1.1A)의 돌입전류가 흐르게 된다.

접점용량 1A로는 용량이 부족되고 접점의 수명을 단축시키게 되는데 이 경우 출력기기(접촉기 코일과 램프)를 병렬로 접속시킬 수는 없고 2개의 출력단자(예를 들면 "012"와 "013")를 사용하면 수명은 그대로 된다. 만약 접점용량이 2A이면 병렬로 접속해서 2개의 기기를 동시에 구동할 수 있다.

4.2 正轉—逆轉運轉回路

저압 3상 유도전동기의 正逆運轉은 그림 15와 같이 2대의 電磁接觸器를 사용하며 3相中 2개의 전선을 바꾸는 것은 잘 알려진 사실이다. 이러한 가역운전회로는 릴레이 시퀀스에서나 PC에 의한



<그림 14> 돌입전류

시퀀스에서도 근본적으로 변함이 없으나 특히 주의할 것은 그림16과 같이 主回路의 電源相間短絡이다.

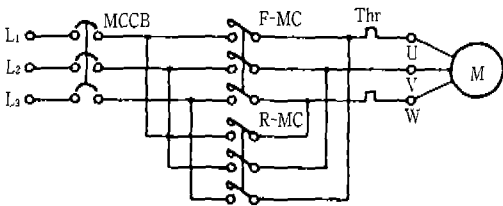
주회로에서는 F-MC와 R-MC의 동시 투입에 의한 상간단락과 片側開路時 아크에 의한 아크단락이 있다.

아크단락이란 예를 들면 F-MC의 주점점이 開路하여 아크가 발생하고 있는 동안(전기적으로 도통하고 있는 동안)에 R-MC의 주점점이 폐회로로 되어 주회로에서 단락하는 것을 말한다.

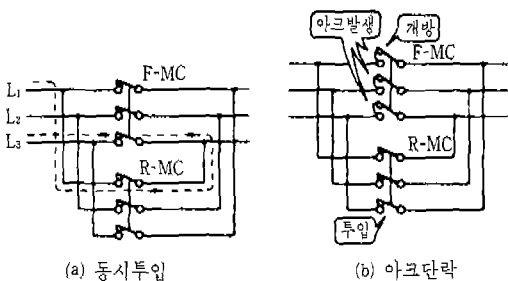
이상의 2개 상간단락을 방지하는 데는 릴레이 시퀀스와는 약간 다른 대책이 필요한데 그림17의 제어회로를 예로 하여 동시 투입방지를 위한 인터록(Interlock)을 설명한다.

그림17 (a)의 회로에서 그림(b)의 I/O 배선을 상상하는 것은 다음에 설명하기로 하고 확실한 투입방지를 위해서는 프로그램내에서 인터록을 취할 수 있는데 이것을 일명 소프트 인터록이라고 한다.

그러나 PC 내부에서 잡음 등에 의한 연습처리



<그림15> 3相 誘導電動機의 可逆主回路



<그림16> 主回路 相間短絡

착오가 있는 경우에는 F-MC와 R-MC를 동시에 출력(ON)하게 된다.

동시투입을 방지하기 위해서는 그림18과 같이 PC 외부(출력측)에서 상호 b점점에 의한 인터록(하드인터록이라고도 한다)을 조성하는 것이 안전하다.

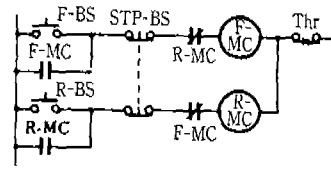
그러나 하드인터록만으로는 소프트인터록이 없는 경우 接點溶着 등과 같이 외부사고일 때는 PC 내부에서 동시에 출력을 ON하게 된다.

그러므로 그림18 (b)와 같이 PC 내부에서는 소프트인터록을, 그리고 외부에서는 하드인터록을 조성할 필요가 있다.

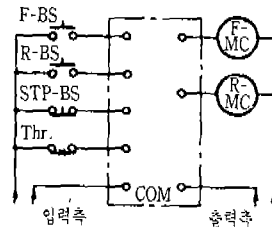
또한 Thr은 PC의 외부에서 처리하고 있기 때문에 이미 설명한 바와 같이 응답신호의 점점에서 출력상태를 확인함과 동시에 Thr이 작동하는 경우에는 PC 내부의 프로그램에 의한 自己維持를 제거할 필요가 있다.

응답신호의 점점은 正轉用(F-MC) 또는 逆轉用(R-MC) 電磁接觸器의 보조 a점점을 입력측에 배선하여 切换시키고 각각의 전자점속기의 동작을 PC에 도입하는 결과가 되므로 동작시간을 고려하여야 한다.

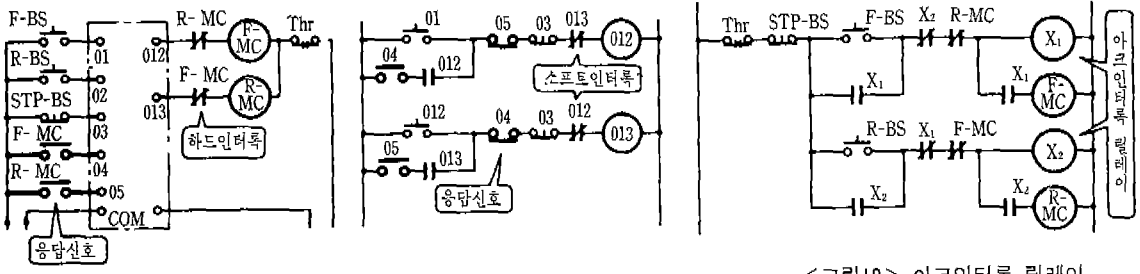
예를 들면 정회전일 때는 F-B-S를 눌러서 PC



(a) 시퀀스



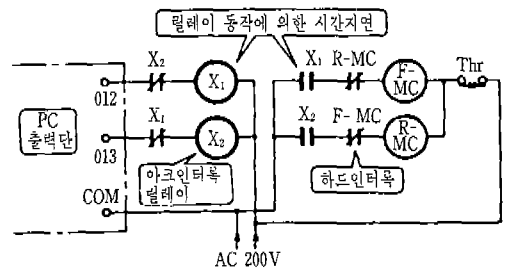
(b) I/O 배선
<그림17> 可逆回路



(a) I/O 배선 (b) 시퀀스
 <그림18> 신뢰성 향상

<그림19> 아크인터록 릴레이

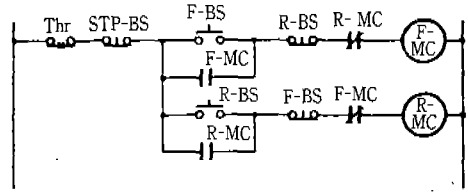
내부의 연산처리로 출력단자 “012”를 ON시킨다.
 이어서 코일 F-MC가 여자되고 그 결과 모터를 정회전시킴과 동시에 PC가 보조 a점점의 폐회로를 도입한 후에 PC내의 프로그램에 自己維持가 걸린다는 것이다.
 한편 응답신호의 접점을 접속한 PC 제어에서는 각각의 전자접촉기가 완전하게 동작하는 것을 확인한 다음에 버튼스위치를 떼어야 자기유지가 된다는 것이다.



<그림20> 외부 아크인터록

다음에 아크단락을 방지하기 위한 아크인터록에 대하여 설명한다.

릴레이 시퀀스에서도 그림19와 같이 아크인터록을 설치하는 예가 있다. 이것은 보조 릴레이 1개를 추가하여 코일여자, 점점동작(폐회로)이라는 릴레이의 동작지연으로서 아크가 소멸할 때까지 시간지연을 만들어 내는 것이다.



<그림21> 직접 전환회로

릴레이 시퀀스에서는 가능한 아크인터록이 될 수 있으나 이것을 그대로 PC로 바꾸어 넣을 수는 없게 된다.

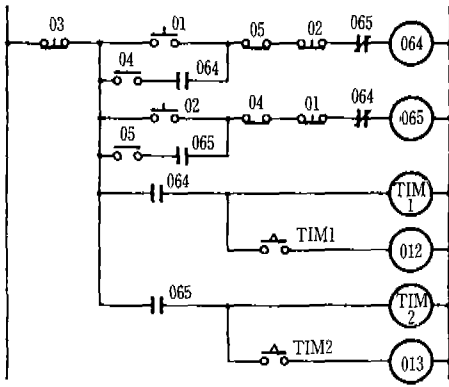
다만 그림20과 같이 외부에 아크인터록 릴레이를 설치하면 릴레이 시퀀스와 같은 효과가 있게 된다. 그러나 버튼스위치로 직접 정역전을 전환할 수 있는 그림21과 같은 회로에서 외부에 아크인터록을 넣지 않는 경우에는 그림22와 같이 프로그램으로 강제로 시간을 지연시켜야 한다.

그림22는 아크시간을 고려하여 타이머에서 강제적으로 출력을 지연시키는 예이다.

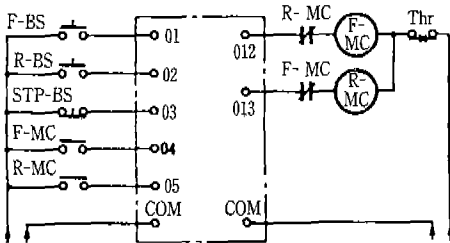
한편 전동기의 운전과 정지에 PC가 사용될 때는 일반적으로 어떠한 자동운전이 이루어진다. PC가 다운(Down)되었을 때나 보수시 수동으로 운전해야 하는 경우 그림23과 같이 PC 외부에서 수동 운전회로가 필요한데 이 때에 전원의 극성에 주의하여야 한다.

4.3 스타-델타시동

스타-델타시동은 3상 유도전동기의 減電壓始



(a) 시퀀스



(b) I/O 배선

<그림22> 가역운전

동에 대한 대표적인 것으로 모터의 고정자 권선을 스타접속으로 하여 시동전류를 전압壓始動의 1/3로 억제해서 시동하는 것이다.

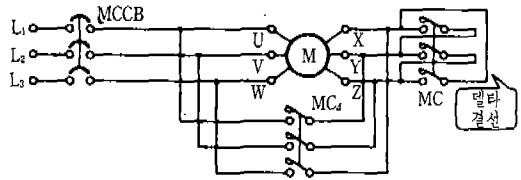
시동후 가속하여 정격속도, 정격전류에 가까워진 곳에서 고정자 권선을 델타접속으로 전환하여 운전하는 방법이다.

스타-델타시동법에는 여러 가지가 있으며 주 회로에는 2콘덕터방식(그림24), 3콘덕터방식(그림25), 오픈 트랜지션방식 및 클로즈 트랜지션방식 등이 있다.

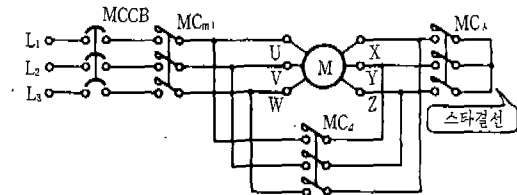
스타용 電磁接觸器의 2차측 결선방법에도 스타 결선과 델타결선이 있으며 델타결선이 델타용 전자접촉기보다 한층 아래에 있는 정격의 스타용 전자접촉기를 선정할 수 있는데 이것은 릴레이 시퀀스의 노하우라고 말할 수 있다.

여기서 스타-델타시동을 PC로 바꾸어 놓을 때 주의사항을 보면 2콘덕터 방식이든 3콘덕터 방식이든 그림26과 같은 主回路的 相間短絡이다.

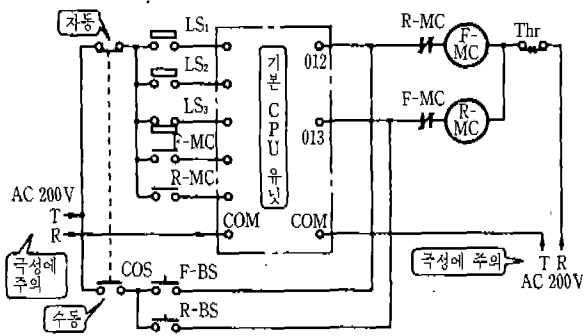
즉 可逆運轉回路에서 설명한 바와 같이 스타용



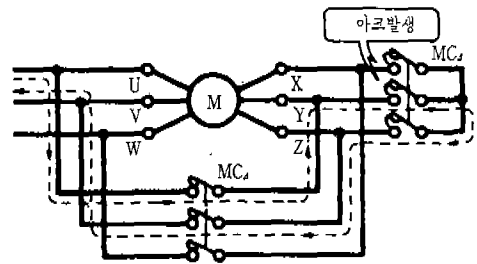
<그림24> 2콘덕터 방식 λ - Δ 시동



<그림25> 3콘덕터 방식 λ - Δ 시동



<그림23> 외부 수동회로



<그림26> 주회로 상간단락

과 델타용 접촉기의 동시투입과 아크단락에 의한 주회로의 상간단락이다.

릴레이 시퀀스에서는 전자접촉기 등의 동작 지연에 의한 시간지연을 잘 이용하면서 이에 대한 대책을 취하고 있으나 PC는 프로그램의 처리속도가 지나치게 빠르므로 릴레이 시퀀스를 직접 바꾸어 놓을 수 없는 일이 많다.

그림27은 릴레이 시퀀스의 스타-델타시동회로에 대한 예로서 여기에는 스타-델타 타이머와 아크인터록 릴레이가 사용되고 있다.

스타-델타 타이머는 일반적인 타이머의 점점 동작과 다르며 그림28과 같이 시간지연을 갖는 b점점이 開路한 다음 다시 일정시간(그림28에서 0.2초)의 지연을 가지고 a점점이 닫히는 타이머이다.

또한 아크인터록 릴레이는 가역운전과 동일하며 그림29와 같이 타이머점점의 폐회로로 코일을 여자하고 점점을 폐회로로 한 후 델타용 접촉기

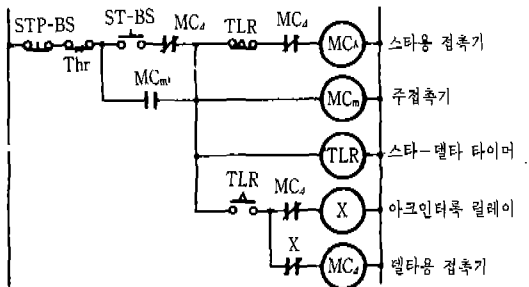
의 코일을 여자하는 것이다.

일반적으로 보조릴레이의 동작시간지연은 20~30ms라고 하며 스타용 접촉기가 투입된 다음 델타용 접촉기가 투입되기까지의 동작순서를 보면 그림30과 같다.

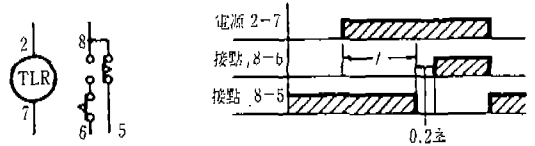
그리고 스타-델타 타이머만으로는 아크의 소멸을 기대하지 못하며 아크단락이 발생하는 경우에는 아크인터록 릴레이(단순한 보조릴레이)가 사용되고 있다.

릴레이 시퀀스에 의한 회로를 PC로 바꾸어 놓으면 그림31과 같은 회로가 되며 I/O 配線은 그림32와 같다.

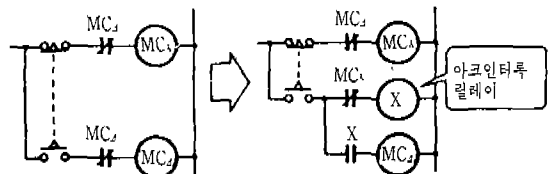
그런데 PC에는 스타-델타 타이머와 같은 기능이 없기 때문에 프로그램의 타이머 TIM1으로 스타시간을 결정하고 아크인터록 릴레이에 상당하



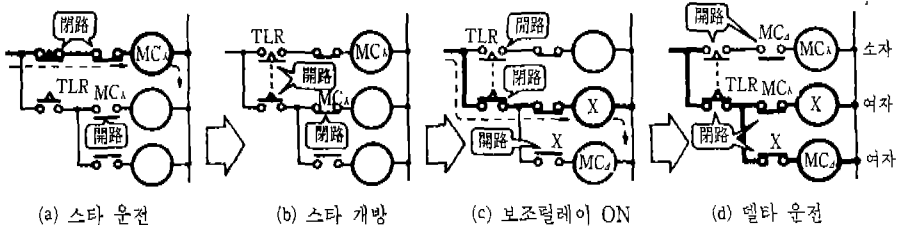
<그림27> 3콘덕터 방식 제어회로



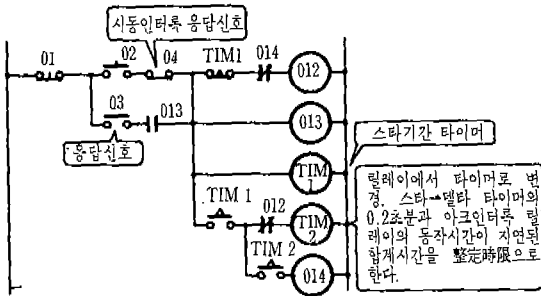
<그림28> 스타-델타 타이머



<그림29> 아크인터록 릴레이



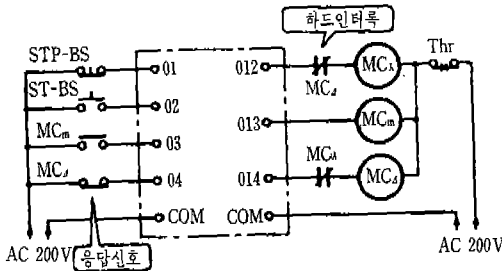
<그림30> 동작순서



<그림31> 스타-델타시동

어드레스	명령	데이터
00	LD	02
01	AND	012
02	OR	00
03	AND	01
04	OUT	012

<그림34> 코딩



<그림32> I/O 배선

는 타이머 TIM2를 여자시킨다.

TIM2의 시간설정은 스타-델타 타이머의 시간 지연(0.2초)과 아크인터록 릴레이의 동작시간지연(0.03초)의 합계시간이 된다.

스타용 접촉기와 델타용 접촉기의 동시투입을 방지하기 위하여 프로그램내에서 소프트인터록과 PC 외부에서의 하드인터록이 필요하다.

델타용 접촉기의 보조 b접점을 시동 인터록을 위하여 응답신호의 접점으로 도입하고 시동 인터록은 어떠한 사고(接點熔着 등)인 경우에도 시동

용 지령(버튼스위치)을 내더라도 시동상태가 되지 않도록 하기 위한 것이다.

또한 주접촉기 MCm의 응답신호 접점은 Thr이 외부에서 작동하였을 때 프로그램내에서自己維持를 제거하기 위한 접점이다.

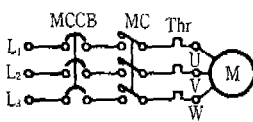
4.4 코딩으로 회로작성

여기서는 3상 유도전동기의 전전압 시동의 예로서 I/O 배선과 코딩시트로 PC용 시퀀스회로를 작성하기로 한다.

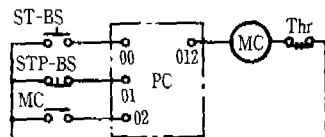
그림33에 主回路와 I/O 배선을 그림34처럼 코딩을 하면 그림35와 같은 회로가 된다.

어드레스에 따라 명령어를 순번으로 배열하는 것이 중요하나 AND와 OR은 그 때까지의 연산 결과에 대한 AND와 OR이라는 것을 주의하여야 한다.

그림36의 주회로는 그림33 (a)와 동일하며 I/O 배선이 다른 예를 나타내고 있다. 그림33 (b)의 코딩에

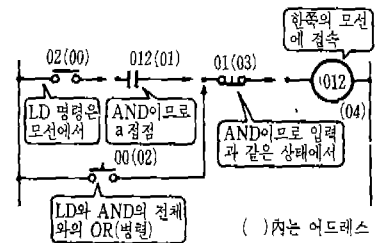


(a) 주회로

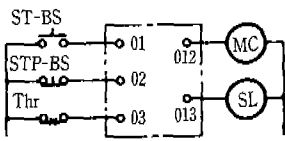


(b) I/O 배선

<그림33> 전전압 시동 (1)



<그림35> 시퀀스회로

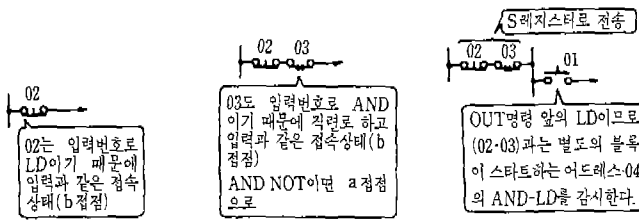


(a) I/O 배선

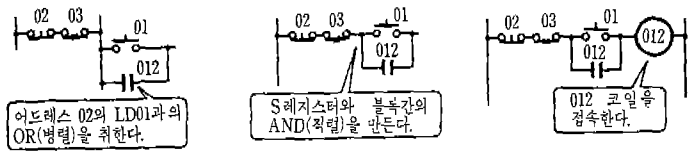
어드레스	명령	데이터
00	LD	02
01	AND	03
02	LD	01
03	OR	012
04	AND LD	-
05	OUT	012
06	LD	012
07	OUT	013

(b) 코딩

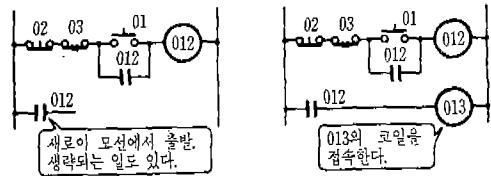
<그림36> 전전압 시동 (2)



(a) 어드레스 00(LD02) (b) 어드레스 01(AND03) (c) 어드레스 02(LD01)



(d) 어드레스 03(OR012) (e) 어드레스 04(AND·LD) (f) 어드레스 05(OUT012)



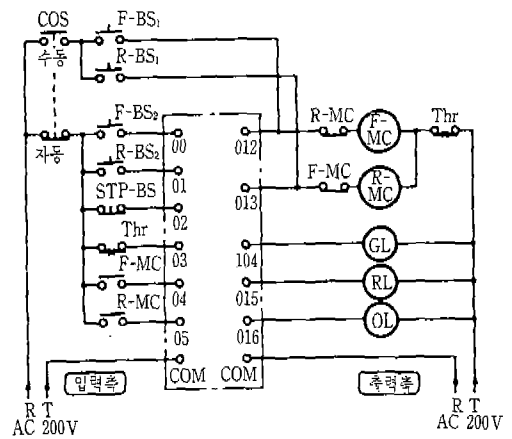
(g) 어드레스 06(LD012) (h) 어드레스 07(OUT013)

<그림37> 작성순서

어드레스	명령	데이터
18	AND NOT	00
19	AND NOT	04
20	AND NOT	012
21	OUT	065
22	TIM 2	005
23	LD TIM	2
24	OUT	013
25	IL END	-
26	LD NOT	012
27	AND NOT	013
28	OUT	014
29	LD	012
30	AND NOT	013
31	OUT	015
32	LD NOT	012
33	AND	013
34	OUT	016
35	END	-

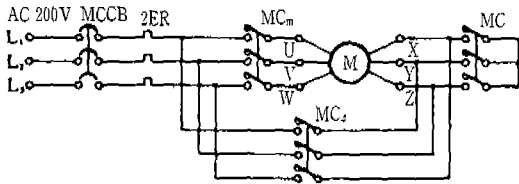
어드레스	명령	데이터
00	LD	02
01	AND	03
02	IL	-
03	LD	00
04	LD	04
05	AND	064
06	OR LD	-
07	AND NOT	01
08	AND NOT	05
09	AND NOT	013
10	OUT	064
11	TIM 1	005
12	LD TIM	1
13	OUT	012
14	LD	01
15	LD	05
16	AND	065
17	OR LD	-

(a) 코딩

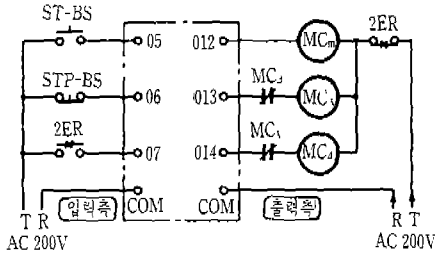


(b) I/O 배선

<그림38> 가역운전회로



(a) 주회로



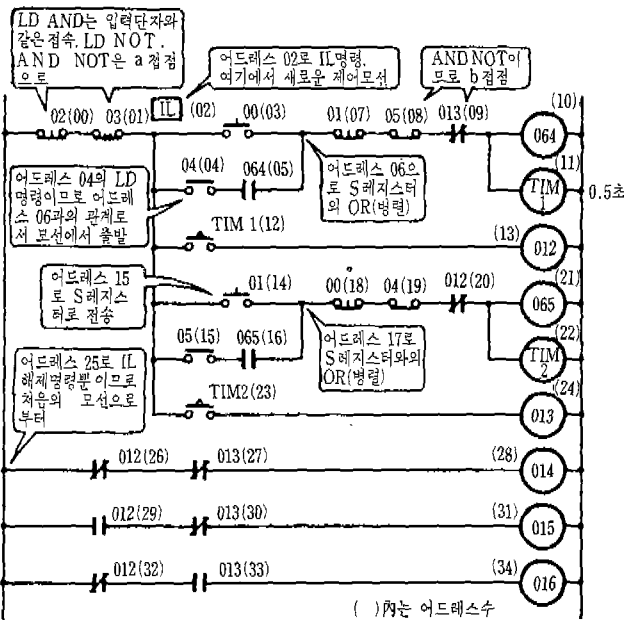
(b) I/O 배선

어드레스	명령	데이터
00	LD	05
01	AND NOT	014
02	OR	064
03	AND	06
04	AND	07
05	OUT	064
06	TIM 1	050
07	LD NOT TIM	1
08	OR TIM	3
09	AND	064
10	OUT	012
11	LD NOT TIM	1

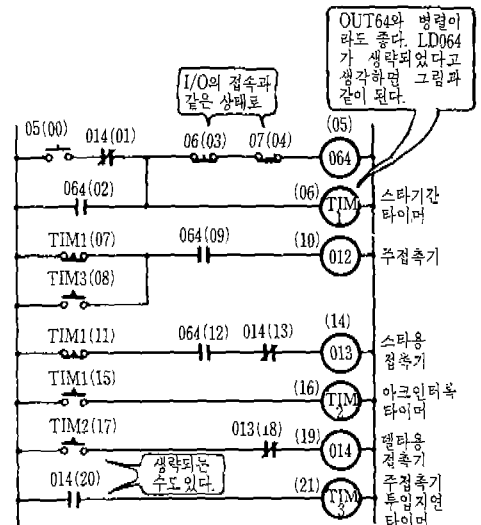
어드레스	명령	데이터
12	AND	064
13	AND NOT	014
14	OUT	013
15	LD TIM	1
16	TIM 2	003
17	LD TIM	2
18	AND NOT	013
19	OUT	014
20	LD	014
21	TIM 3	002
22	END	-

(c) 코딩

<그림 40> 스타-델타시동



<그림 39> 시퀀스



<그림 41> 시퀀스

따라서 작성한 회로를 그림37에 나타내었다.

어드레스 "02"에서 별도로 논리스타트 명령 (LD)이 출력되고 있는데 이 경우에는 S 레지스터와 직렬인가 병렬인가를 어드레스를 보면서 그리고 OR,LD와 AND,LD에 주의하면서 회로도를 작성하여야 한다.

끝으로 그림11을 참고하여 I/O 배선과 시퀀스 도의 관계에서 a 점인가 b점점인가를 확인할 필요가 있다.

그림38과 그림39는 가역운전회로의 예를 그리고 스타-델타시동 회로의 예를 그림40과 그림41에 나타내었다.

▣ 다음 호에 계속