

项目 2

简单电动机控制电路的 PLC 控制

2.1 相关知识

2.1.1 可编程控制器程序设计语言

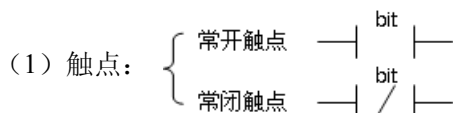
在可编程控制器中有多种程序设计语言，它们是梯形图、语句表、顺序功能流程图、功能块图等。

梯形图和语句表是基本程序设计语言，它通常由一系列指令组成，用这些指令可以完成大多数简单的控制功能，例如，代替继电器、计数器、计时器完成顺序控制和逻辑控制等，通过扩展或增强指令集，它们也能执行其他的基本操作。

供 S7-200 系列 PLC 使用的 STEP 7-Micro/WIN 编程软件支持 SIMATIC 和 IEC1131-3 两种基本类型的指令集，SIMATIC 是 PLC 专用的指令集，执行速度快，可使用梯形图、语句表、功能块图编程语言。IEC1131-3 是可编程控制器编程语言标准，但指令集中指令较少，只能使用梯形图和功能块图两种编程语言。SIMATIC 指令集的某些指令不是 IEC1131-3 中的标准指令。SIMATIC 指令和 IEC1131-3 中的标准指令系统并不兼容。我们将重点介绍 SIMATIC 指令。

1. 梯形图（Ladder Diagram）程序设计语言

梯形图程序设计语言是最常用的一种程序设计语言。它来源于继电器逻辑控制系统的描述。在工业过程控制领域，电气技术人员对继电器逻辑控制技术较为熟悉，因此，由这种逻辑控制技术发展而来的梯形图受到了欢迎，并得到了广泛的应用。梯形图与操作原理图相对应，具有直观性和对应性；与原有的继电器逻辑控制技术的不同点是：梯形图中的能流不是实际意义的电流，内部的继电器也不是实际存在的继电器，因此，应用时，需与原有继电器逻辑控制技术的有关概念区别对待。LAD 图形指令有 3 个基本形式：



触点符号代表输入条件如外部开关、按钮及内部条件等。CPU 运行扫描到触点符号时，到触点位指定的存储器位访问（即 CPU 对存储器的读操作）。该位数据（状态）为 1 时，表示“能流”能通过。计算机读操作的次数不受限制，用户程序中，常开触点、常闭触点可以使用无数次。

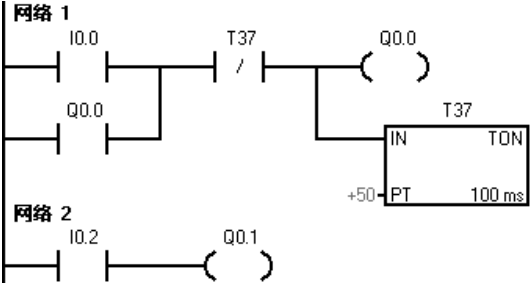
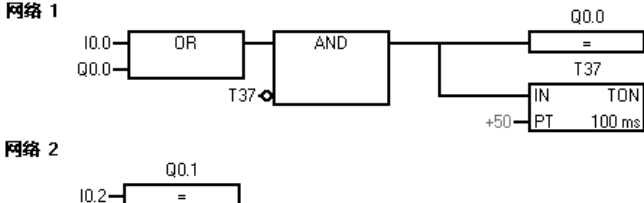
(2) 线圈：——(^{bit})

线圈表示输出结果，通过输出接口电路来控制外部的指示灯、接触器等及内部的输出条件等。线圈左侧接点组成的逻辑运算结果为 1 时，“能流”可以达到线圈，使线圈得电动作，CPU 将线圈的位地址指定的存储器的位置位为 1；逻辑运算结果为 0，线圈不通电，存储器的位置 0。即线圈代表 CPU 对存储器的写操作。PLC 采用循环扫描的工作方式，所以在用户程序中，每个线圈只能使用一次。

(3) 指令盒：指令盒代表一些较复杂的功能。如定时器，计数器或数学运算指令等。当“能流”通过指令盒时，执行指令盒所代表的功能。

梯形图按照逻辑关系可分成网络段，分段只是为了阅读和调试方便。在本书部分举例中我们将网络段省去。表 2-1 是梯形图示例。

表 2-1 梯形图与语句表

梯形图	语句表
	<pre> 网络 1 LD I0.0 O Q0.0 AN T37 = Q0.0 TON T37, +50 网络 2 LD I0.2 = Q0.1 </pre>
功能块图	
	

2. 语句表（Statement List）程序设计语言

语句表程序设计语言是用布尔助记符来描述程序的一种程序设计语言。语句表程序设计语言与计算机中的汇编语言非常相似，采用布尔助记符来表示操作功能。

语句表程序设计语言具有下列特点：

- (1) 采用助记符来表示操作功能，具有容易记忆、便于掌握的特点。
- (2) 在编程器的键盘上采用助记符表示，具有便于操作的特点，可在无计算机的场合进行编程设计。

(3) 用编程软件可以将语句表与梯形图相互转换，如表 2-1 所示。

3. 功能块图（Function Block Diagram, FBD）程序设计语言

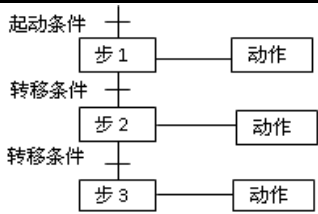
功能块图程序设计语言是采用逻辑门电路的编程语言，有数字电路基础的人很容易掌握。功能块图指令由输入、输出段及逻辑关系函数组成。用 STEP 7-Micro/WIN 编程软件将表 2-1 所示的梯形图转换为 FBD 程序，如表 2-1 所示。方框的左侧为逻辑运算的输入变量，右侧为

输出变量，输入输出端的小圆圈表示“非”运算，信号自左向右流动。

4. 顺序功能流程图（Sequential Function Chart）程序设计

顺序功能流程图程序设计是近年来发展起来的一种程序设计。采用顺序功能流程图的描述，控制系统被分为若干个子系统，从功能入手，使系统的操作具有明确的含义，便于设计人员和操作人员设计思想的沟通，便于程序的分工设计和检查调试。顺序功能流程图的主要元素是步、转移、转移条件和动作。顺序功能流程图及特点如表 2-2 所示。

表 2-2 顺序功能流程图

顺序功能流程图	特点
	①以功能为主线，条理清楚，便于对程序操作的理解和沟通； ②对大型的程序，可分工设计，采用较为灵活的程序结构，可节省程序设计时间和调试时间； ③常用于系统规模较大，程序关系较复杂的场合； ④只有在活动步的命令和操作被执行后，才对活动步后的转换进行扫描，因此，整个程序的扫描时间要大大缩短

2.1.2 基本位操作指令

位操作指令是 PLC 常用的基本指令，梯形图指令有触点和线圈两大类，触点又分常开触点和常闭触点两种形式；语句表指令有与、或及输出等逻辑关系，位操作指令能够实现基本的位逻辑运算和控制。

1. 逻辑取（装载）及线圈驱动指令 LD/LDN/=

（1）指令功能：

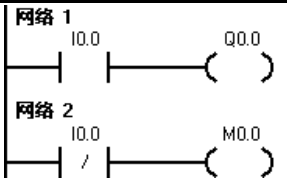
LD（Load）：常开触点逻辑运算的开始。对应梯形图则为在左侧母线或线路分支点处初始装载一个常开触点。

LDN（Load not）：常闭触点逻辑运算的开始（即对操作数的状态取反），对应梯形图则为在左侧母线或线路分支点处初始装载一个常闭触点。

=（OUT）：输出指令，对应梯形图则为线圈驱动。对同一元件只能使用一次。

（2）指令格式如表 2-3 所示。

表 2-3 LD/LDN、OUT 指令的使用

梯形图	语句表	操作数
	网络 1 LD I0.0 //装载常开触点 = Q0.0 //输出线圈 网络 2 LDN I0.0 //装载常闭触点 = M0.0 //输出线圈	LD/LDN 的操作数： I、Q、M、SM、T、C、V、S “=”（OUT）的操作数： Q、M、SM、T、C、V、S

说明：

1）触点代表 CPU 对存储器的读操作，常开触点和存储器的位状态一致，常闭触点和存储器的位状态相反。用户程序中同一触点可使用无数次。

如：存储器 I0.0 的状态为 1，则对应的常开触点 I0.0 接通，表示能流可以通过；而对应的常闭触点 I0.0 断开，表示能流不能通过。存储器 I0.0 的状态为 0，则对应的常开触点 I0.0 断开，表示能流不能通过；而对应的常闭触点 I0.0 接通，表示能流可以通过。

2) 线圈代表 CPU 对存储器的写操作，若线圈左侧的逻辑运算结果为“1”，表示能流能够达到线圈，CPU 将该线圈所对应的存储器的位置位为“1”，若线圈左侧的逻辑运算结果为“0”，表示能流不能够达到线圈，CPU 将该线圈所对应的存储器的位写入“0”。用户程序中，同一线圈只能使用一次。

(3) LD、LDN、= 指令使用说明：

LD、LDN 指令用于与输入公共母线（输入母线）相联的接点，也可与 OLD、ALD 指令配合使用于分支回路的开头。

“=” 指令用于 Q、M、SM、T、C、V、S。但不能用于输入映像寄存器 I。输出端不带负载时，控制线圈应尽量使用 M 或其他，而不用 Q。“=” 可以并联使用任意次，但不能串联。如图 2-1 所示。



图 2-1 输出指令可以并联使用

2. 触点串联指令 A/AN

(1) 指令功能：

A (And)：与操作，在梯形图中表示串联连接单个常开触点。

AN (And not)：与非操作，在梯形图中表示串联连接单个常闭触点。

(2) 指令格式如表 2-4 所示。

表 2-4 A/AN 指令格式

梯形图	语句表	
	网络 1	网络 2
	LD I0.0 //装载常开触点	LD Q0.0 //装载常开触点
	A M0.0 //与常开触点	AN I0.1 //与常闭触点
	= Q0.0 //输出线圈	= M0.0 //输出线圈
		A T37 //与常开触点
		= Q0.1 //输出线圈

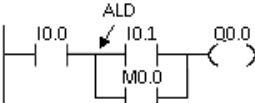
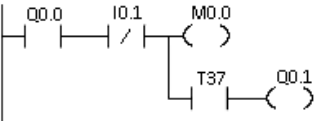
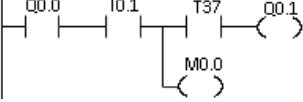
A/AN 的操作数：I、Q、M、SM、T、C、V、S

(3) A/AN 指令使用说明如表 2-5 所示。

表 2-5 A/AN 指令使用说明

A、AN 是单个触点串联连接指令	网络 1	LD M0.0
	M0.0 T37 T38 Q0.0	A T37
		AN T38
		= Q0.0

续表

串联多个接点组合回路时，必须使用 ALD 指令		
若按正确次序编程（即输入：左重右轻、上重下轻；输出：上轻下重），可以反复使用“=”指令		LD Q0.0 AN I0.1 = M0.0 A T37 = Q0.1
若按右图所示的编程次序，就不能连续使用“=”指令		

3. 触点并联指令 O/ON

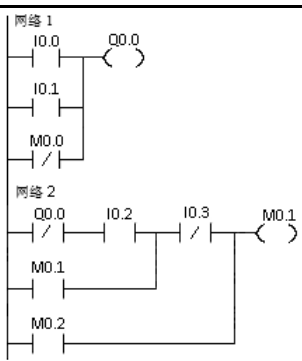
（1）指令功能

O（Or）：或操作，在梯形图中表示并联连接一个常开触点。

ON（Or not）：或非操作，在梯形图中表示并联连接一个常闭触点。

（2）指令格式如表 2-6 所示。

表 2-6 O/ON 指令格式

梯形图	语句表		操作数
	网络 1 LD I0.0 O I0.1 ON M0.0 = Q0.0	网络 2 LDN Q0.0 A I0.2 O M0.1 AN I0.3 O M0.2 = M0.1	ON 操作数： I、Q、M、SM、V、S、T、C

（3）O/ON 指令使用说明：

O/ON 指令可作为并联一个触点指令，紧接在 LD/LDN 指令之后用，即对其前面的 LD/LDN 指令所规定的触点并联一个触点，可以连续使用。

若要并联连接两个以上触点的串联回路时，须采用 OLD 指令。

4. 电路块的串联指令 ALD

（1）指令功能：

ALD：块“与”操作，用于串联连接多个并联电路组成的电路块。

（2）指令格式如表 2-7 所示。

（3）ALD 指令使用说明：

并联电路块与前面电路串联连接时，使用 ALD 指令。分支的起点用 LD/LDN 指令，并联电路结束后使用 ALD 指令与前面电路串联。

可以顺次使用 ALD 指令串联多个并联电路块，支路数量没有限制。如表 2-7 所示。

表 2-7 ALD 指令格式

梯形图		多个并联电路块
语句表	<pre>LD I1.0 //装入常开触点 O I1.1 //或常开触点 LD I1.2 //装入常开触点 O I1.3 //或常开触点 ALD //块与操作 = Q0.0 //输出线圈</pre>	<pre>LD I0.0 ON I0.3 LD I0.1 O I0.4 ALD LD I0.2 O I0.5 ALD = Q0.0</pre>
操作数	ALD 指令无操作数	

5. 电路块的并联指令 OLD

(1) 指令功能:

OLD: 块“或”操作, 用于并联连接多个串联电路组成的电路块。

(2) 指令格式如表 2-8 所示。

表 2-8 OLD 指令格式表

梯形图	语句表	操作数
	<pre>LD I0.0 //装入常开触点 A I0.1 //与常开触点 LD I0.2 //装入常开触点 A I0.3 //与常开触点 OLD //块或操作 = Q0.0 //输出线圈</pre>	OLD 指令无操作数

(3) OLD 指令使用说明:

并联连接几个串联支路时, 其支路的起点以 LD、LDN 开始, 并联结束后用 OLD。

可以顺次使用 OLD 指令并联多个串联电路块, 支路数量没有限制。

【例 2-1】根据图 2-2 所示梯形图, 写出对应的语句表。

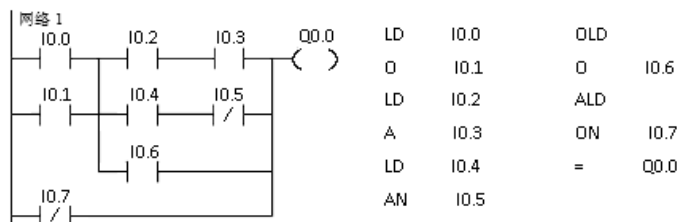


图 2-2 例 2-1 图

6. 逻辑堆栈的操作

S7-200 系列采用模拟栈的结构，用于保存逻辑运算结果及断点的地址，称为逻辑堆栈。
S7-200 系列 PLC 中有一个 9 层的堆栈。在此讨论断点保护功能的堆栈操作。

(1) 指令的功能。堆栈操作指令用于处理线路的分支点。在编制控制程序时，经常遇到多个分支电路同时受一个或一组触点控制的情况，如表 2-9 所示，若采用前述指令不容易编写程序，用堆栈操作指令则可方便地将表 2-9 所示梯形图转换为语句表。

表 2-9 堆栈指令格式

梯形图	语句表	
	LD I0.0 //装载常开触点 LPS //压入堆栈 LD I0.1 //装载常开触点 O I0.2 //或常开触点 ALD //块与操作 = Q0.0 //输出线圈 LRD //读栈	LD I0.3 //装载常开触点 O I0.4 //或常开触点 ALD //块与操作 = Q0.1 //输出线圈 LPP //出栈 A I0.5 //与常开触点 = Q0.2 //输出线圈

LPS（入栈）指令：LPS 指令把栈顶值复制后压入堆栈，栈中原来数据依次下移一层，栈底值压出丢失。

LRD（读栈）指令：LRD 指令把逻辑堆栈第二层的值复制到栈顶，2-9 层数据不变，堆栈没有压入和弹出。但原栈顶的值丢失。

LPP（出栈）指令：LPP 指令把堆栈弹出一级，原第二级的值变为新的栈顶值，原栈顶数据从栈内丢失。

LPS、LRD、LPP 指令的操作过程如表 2-10 所示。图中 ivx 为存储在栈区的断点的地址。

表 2-10 LPS/LRD/LPP 指令的操作过程

LPS 进栈		LRD 读栈		LPP 出栈	
前	后	前	后	前	后
iv0	iv0	iv0	iv0	iv0	iv1
iv1	iv0	iv1	iv1	iv1	iv2
iv2	iv1	iv2	iv2	iv2	iv3
iv3	iv2	iv3	iv3	iv3	iv4
iv4	iv3	iv4	iv4	iv4	iv5
iv5	iv4	iv5	iv5	iv5	iv6
iv6	iv5	iv6	iv6	iv6	iv7
iv7	iv6	iv7	iv7	iv7	iv8
iv8	iv7	iv8	iv8	iv8	X

(2) 指令格式如表 2-9 所示。

(3) 指令使用说明：逻辑堆栈指令可以嵌套使用，最多为 9 层。

为保证程序地址指针不发生错误，入栈指令 LPS 和出栈指令 LPP 必须成对使用，最后一次读栈操作应使用出栈指令 LPP。

堆栈指令没有操作数。

7. 置位/复位指令 S/R

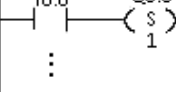
(1) 指令功能

置位指令 S: 使能输入有效后从起始位 S-bit 开始的 N 个位置“1”并保持。

复位指令 R: 使能输入有效后从起始位 S-bit 开始的 N 个位置“0”并保持。

(2) 指令格式及用法如表 2-11 所示。

表 2-11 S/R 指令格式

S/R 指令格式		S/R 使用示例	
梯形图	语句表		
	S S-bit,N		网络 1 LD I0.0 S Q0.0, 1 : 网络 2 LD I0.1 R Q0.0, 1
	R S-bit,N		

(3) 指令使用说明:

对同一元件(同一寄存器的位)可以多次使用 S/R 指令(与“=”指令不同)。

由于是扫描工作方式,当置位、复位指令同时有效时,写在后面的指令具有优先权。

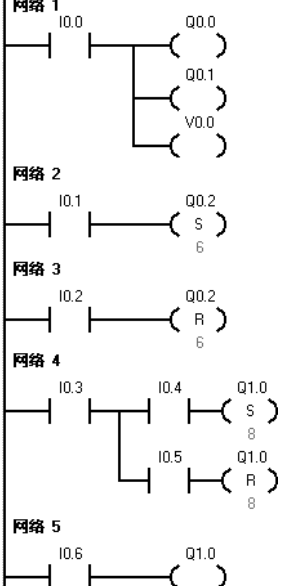
操作数 N 为: VB, IB, QB, MB, SMB, SB, LB, AC, 常量, *VD, *AC, *LD。取值范围为: 0~255。数据类型为: 字节。

操作数 S-bit 为: I, Q, M, SM, T, C, V, S, L。数据类型为: 布尔。

置位复位指令通常成对使用,也可以单独使用或与指令盒配合使用。

【例 2-2】基本位指令举例,如表 2-12 所示。

表 2-12 基本位指令举例

梯形图	语句表
	网络 1 网络标题 LD I0.0 = Q0.0 = Q0.1 = Q0.2 网络 2 //连续将一组6位置为1 //指定起始的位置和置位的个数 LD I0.1 S Q0.2, 6 网络 3 //连续将一组6位置为0 //指定起始的位置和置位的个数 LD I0.2 R Q0.2, 6 网络 4 //置位和复位一组8个输出位 LD I0.3 LPS A I0.4 S Q1.0, 8 LPP A I0.5 R Q1.0, 8 网络 5 //置位和复位指令实现锁存器功能 LD I0.6 = Q1.0

【例 2-3】逻辑堆栈指令程序举例，如表 2-13 所示。

表 2-13 逻辑堆栈指令程序举例

梯形图	语句表																														
	<table><tr><td>网络 1</td><td>网络 2</td></tr><tr><td>LD I0.0</td><td>LD I0.0</td></tr><tr><td>LD I0.1</td><td>LPS</td></tr><tr><td>LD I2.0</td><td>LD I0.5</td></tr><tr><td>A I2.1</td><td>O I0.6</td></tr><tr><td>OLD</td><td>ALD</td></tr><tr><td>ALD</td><td>= Q7.0</td></tr><tr><td>= Q0.0</td><td>LRD</td></tr><tr><td></td><td>LD I2.1</td></tr><tr><td></td><td>O I1.3</td></tr><tr><td></td><td>ALD</td></tr><tr><td></td><td>= Q6.0</td></tr><tr><td></td><td>LPP</td></tr><tr><td></td><td>A I1.0</td></tr><tr><td></td><td>= Q3.0</td></tr></table>	网络 1	网络 2	LD I0.0	LD I0.0	LD I0.1	LPS	LD I2.0	LD I0.5	A I2.1	O I0.6	OLD	ALD	ALD	= Q7.0	= Q0.0	LRD		LD I2.1		O I1.3		ALD		= Q6.0		LPP		A I1.0		= Q3.0
网络 1	网络 2																														
LD I0.0	LD I0.0																														
LD I0.1	LPS																														
LD I2.0	LD I0.5																														
A I2.1	O I0.6																														
OLD	ALD																														
ALD	= Q7.0																														
= Q0.0	LRD																														
	LD I2.1																														
	O I1.3																														
	ALD																														
	= Q6.0																														
	LPP																														
	A I1.0																														
	= Q3.0																														

8. RS 触发器指令

RS 触发器指令包括置位优先触发指令（SR）和复位优先触发指令（RS）。

置位优先触发器是一个置位优先的锁存器，当置位信号（S1）和复位信号（R）都为 1 时，输出为 1；复位优先触发器是一个复位优先的锁存器，当置位信号（S）和复位信号（R1）都为 1 时，输出为 0。Bit 参数用于指定被置位或者复位的位元件。

RS 触发器指令梯形图符合及反映指令功能的真值表如表 2-14 所示。RS 触发器指令的有效操作数如表 2-15 所示。

表 2-14 RS 触发器指令及真值表

指令 SR		SI	R	输出（bit）
置位优先 触发指令		0	0	保持前一状态
		0	1	0
		1	0	1
		1	1	1
指令 RS		S	R1	输出（bit）
复位优先 触发指令		0	0	保持前一状态
		0	1	0
		1	0	1
		1	1	0

表 2-15 RS 触发器指令的有效操作数

输入/输出	数据类型	操作数
S1、R	BOOL	I、O、V、M、SM、S、T、C、能流
S、R1、OUT	BOOL	I、O、V、M、SM、S、T、C、L、能流
Bit	BOOL	I、O、V、M、S

【例 2-4】RS 触发器指令程序举例，如表 2-16 所示。

表 2-16 RS 触发器指令程序举例

梯形图	时序图

2.1.3 定时器指令

S7-200 系列 PLC 的定时器是对内部时钟累计时间增量计时的。每个定时器均有一个 16 位的当前值寄存器用以存放当前值（16 位符号整数）；一个 16 位的预置值寄存器用以存放时间的设定值；还有一位状态位，反应其触点的状态。

1. 工作方式

S7-200 系列 PLC 定时器按工作方式分三大类定时器。其指令格式如表 2-17 所示。

表 2-17 定时器的指令格式

梯形图	语句表	说明
	TON T××, PT	TON—通电延时定时器 TONR—记忆型通电延时定时器 TOF—断电延时定时器
	TONR T××, PT	IN 是使能输入端，指令盒上方输入定时器的编号（T××），范围为 T0-T255；PT 是预置值输入端，最大预置值为 32767；PT 的数据类型：INT；
	TOF T××, PT	PT 操作数有：IW, QW, MW, SMW, T, C, VW, SW, AC, 常数

2. 时基

按时基脉冲分，则有 1ms、10ms、100ms 三种定时器。采用不同的时基标准，定时精度、定时范围和定时器刷新的方式也不同。

(1) 定时精度和定时范围

定时器的工作原理是：使能输入有效后，当前值 PT 对 PLC 内部的时基脉冲增 1 计数，当计数值大于或等于定时器的预置值后，状态位置 1。其中，最小计时单位为时基脉冲的宽度，又为定时精度；从定时器输入有效，到状态位输出有效，经过的时间为定时时间，即：定时时间=预置值×时基。当前值寄存器为 16bit，最大计数值为 32767，由此可推算不同分辨率的定时器的设定时间范围。CPU22X 系列 PLC 的 256 个定时器分属 TON（TOF）和 TONR 工作方式，以及 3 种时基标准，如表 2-18 所示。可见时基越大，定时时间越长，但精度越差。

表 2-18 定时器的类型

工作方式	时基 (ms)	最大定时范围 (s)	定时器号
TONR	1	32.767	T0, T64
	10	327.67	T1-T4, T65-T68
	100	3276.7	T5-T31, T69-T95
TON/TOF	1	32.767	T32, T96
	10	327.67	T33-T36, T97-T100
	100	3276.7	T37-T63, T101-T255

(2) 1ms、10ms、100ms 定时器的刷新方式不同

1ms 定时器每隔 1ms 刷新一次，与扫描周期和程序处理无关，即采用中断刷新方式。因此当扫描周期较长时，在一个周期内可能被多次刷新，其当前值在一个扫描周期内不一定保持一致。

10ms 定时器则由系统在每个扫描周期开始自动刷新。由于每个扫描周期内只刷新一次，故而每次程序处理期间，其当前值为常数。

100ms 定时器则在该定时器指令执行时刷新。下一条执行的指令，即可使用刷新后的结果，非常符合正常的思路，使用方便可靠。但应当注意，如果该定时器的指令不是每个周期都执行，定时器就不能及时刷新，可能导致出错。

3. 定时器指令工作原理

下面我们将从原理应用等方面分别叙述通电延时型、有记忆的通电延时型、断电延时型三种定时器的使用方法。

(1) 通电延时定时器（TON）指令工作原理

程序及时序分析如表 2-19 所示。当 I0.0 接通时即使能端（IN）输入有效时，驱动 T37 开始计时，当前值从 0 开始递增，计时到设定值 PT 时，T37 状态位置 1，其常开触点 T37 接通，驱动 Q0.0 输出，其后当前值仍增加，但不影响状态位。当前值的最大值为 32767。当 I0.0 分断时，使能端无效，T37 复位，当前值清 0，状态位也清 0，即回复原始状态。若 I0.0 接通时间未到设定值就断开，T37 则立即复位，Q0.0 不会有输出。

(2) 记忆型通电延时定时器（TONR）指令工作原理

使能端（IN）输入有效（接通）时，定时器开始计时，当前值递增，当前值大于或等于预置值（PT）时，输出状态位置 1。使能端输入无效（断开）时，当前值保持（记忆），使能端（IN）再次接通有效时，在原记忆值的基础上递增计时。

表 2-19 通电延时定时器（TON）指令应用示例

梯形图	语句表
	NETWORK 1 // 在 (10) 100 毫秒或 1 秒之后, 100 毫秒定时器 T37 超时 // I0.0 闭合=T37 被启用, I0.0 断开=禁止和复原 T37 LD I0.0 TON T37 +10 NETWORK 2 // T37 位由定时器 T37 控制 LD T37 = Q0.0
时序图	

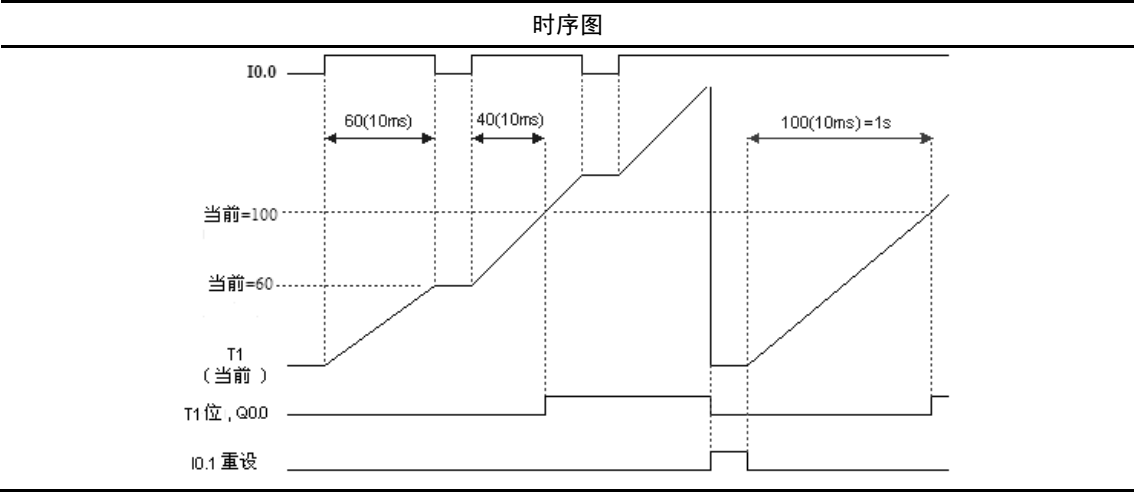
注意: TONR 记忆型通电延时定时器采用线圈复位指令 R 进行复位操作, 当复位线圈有效时, 定时器当前位清零, 输出状态位置 0。

程序分析如表 2-20 所示。如 T1, 当输入 IN 为 1 时, 定时器计时; 当 IN 为 0 时, 其当前值保持并不复位; 下次 IN 再为 1 时, T1 当前值从原保持值开始往上加, 将当前值与设定值 PT 比较, 当前值大于等于设定值时, T1 状态位置 1, 驱动 Q0.0 有输出, 以后即使 IN 再为 0, 也不会使 T1 复位, 要使 T1 复位, 必须使用复位指令。

表 2-20 记忆型通电延时定时器（TONR）指令应用示例

梯形图	语句表
	NETWORK 1 //10 毫秒 TONR 定时器在 PT= (100×10 毫秒) 或 1 秒时超时 LD I0.0 TONR T1 +100 NETWORK 2 //T1 位由定时器 T1 控制 //在定时器总共累积 1 秒后, 打开 Q0.0 LD T1 = Q0.0 NETWORK 3 //TONR 定时器必须由带有 T 地址的复原指令复原 //当 I0.1 打开时, 复原定时器 T1 (当前和位) LD I0.1 R T1 1

续表



(3) 断电延时型定时器 (TOF) 指令工作原理

断电延时型定时器用来在输入断开, 延时一段时间后, 才断开输出。使能端 (IN) 输入有效时, 定时器输出状态位立即置 1, 当前值复位为 0。使能端 (IN) 断开时, 定时器开始计时, 当前值从 0 递增, 当前值达到预置值时, 定时器状态位复位为 0, 并停止计时, 当前值保持。

如果输入断开的的时间小于预定时间, 定时器仍保持接通。IN 再接通时, 定时器当前值仍设为 0。断电延时定时器的应用程序及时序分析如表 2-21 所示。

表 2-21 断电延时型定时器 (TOF) 指令应用示例

梯形图 (LAD)	语句表 (STL)
Network 1 Network 2	NETWORK 1 // 在 (100×10 毫秒) 或 1 秒之后, 10 毫秒定时器 T33 超时 // I0.0 闭合至断开=T33 被启用, I0.0 断开至闭合=禁止和复原 T33 LD I0.0 TOF T33 +100 NETWORK 2 // 定时器 T33 通过定时器触点 T33 控制 Q0.0 // LD T33 = Q0.0
时序图	

2.2 项目实施

2.2.1 电动机单向运行的 PLC 控制

1. I/O 配置（见表 2-22）与控制电路

表 2-22 PLC 的 I/O 配置

序号	类型	设备名称	信号地址	编号
1	输入	停止按钮	I0.0	SB1
2		起动按钮	I0.1	SB2
3		过载保护	I0.2	FR
4	输出	接触器	Q0.0	KM

2. 输入/输出接线

PLC 的 I/O 接线图如图 2-3 所示。

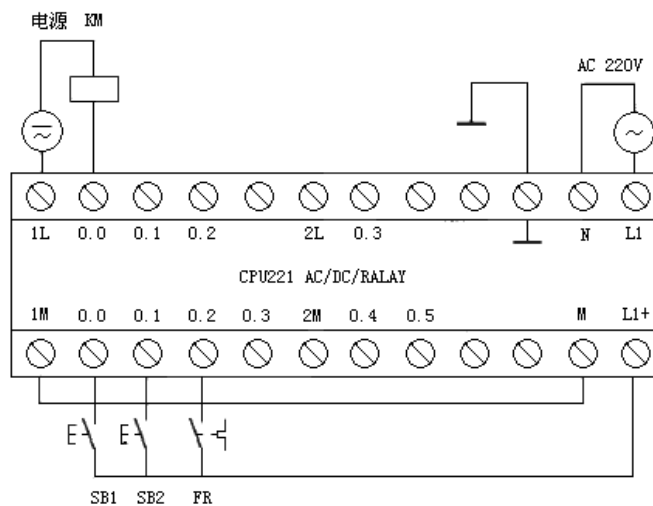


图 2-3 PLC 的 I/O 接线图

3. 梯形图程序设计（见图 2-4）

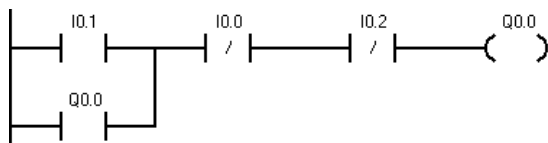


图 2-4 梯形图

2.2.2 自动循环控制电路的 PLC 控制

1. 自动往返继电器 - 接触器控制线路（如图 2-5 所示）

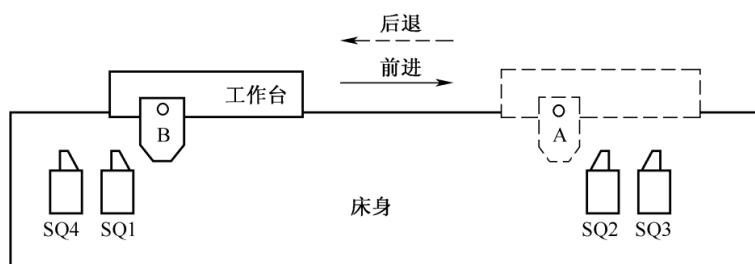
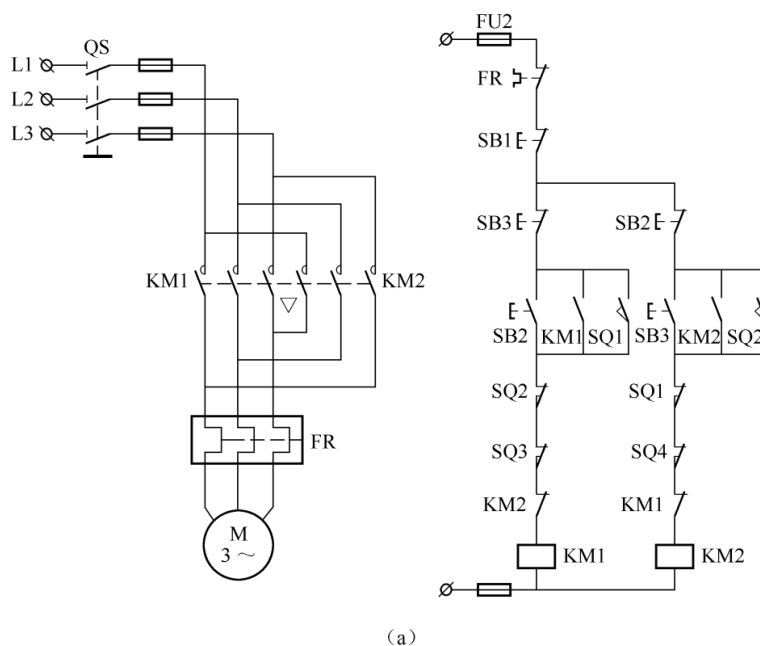


图 2-5 自动往返继电—接触器控制线路

2. 自动往返 PLC 控制的 I/O 配置 (见表 2-23) 与控制电路

表 2-23 PLC 的 I/O 配置

序号	类型	设备名称	信号地址	编号
1	输入	停止按钮	I0.0	SB1
2		正转起动按钮	I0.1	SB2
3		反转起动按钮	I0.2	SB3
4		过载保护	I0.3	FR
5		正转限位	I0.4	SQ2
6		反转限位	I0.5	SQ1
7		正转极限	I0.6	SQ3
8		反转极限	I0.7	SQ4
9	输出	电动机正转接触器	Q0.0	KM1
10		电动机正转接触器	Q0.1	KM2

3. 自动往返 PLC 控制的输入/输出接线(如图 2-6 所示)

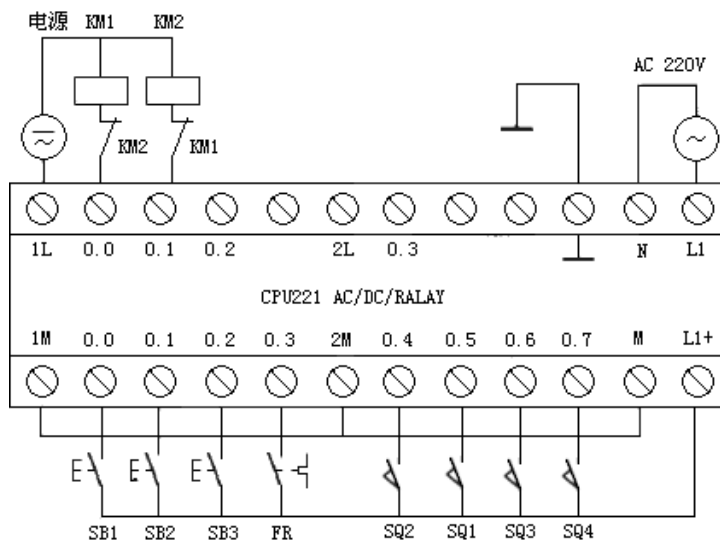


图 2-6 PLC 的 I/O 接线图

4. 自动往返 PLC 控制的梯形图程序设计(如图 2-7 所示)

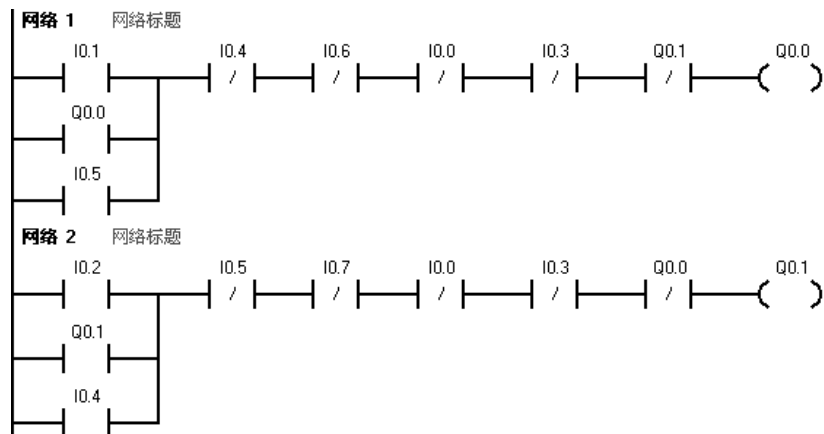


图 2-7 梯形图

2.2.3 电动机正反转 Y- Δ 降压起动的 PLC 控制

1. Y- Δ 降压起动继电器-接触器控制线路

有些生产机械设备,要求加工时采用 Y- Δ 降压起动方式的三相鼠笼型电动机来拖动。

如图 2-8 所示。该电路能实现 Y- Δ 降压起动过程。三相鼠笼型电动机由 KM_{Δ} 和 KM_Y 两个接触器来控制 Y- Δ 转换。

工作原理如下:按下起动按钮 SB1,KT、 KM_Y 、KM 通电并自保,电动机接成 Y 形起动,3s 后,KT 延时断开的常闭触点动作,使 KM_Y 断电, KM_{Δ} 通电吸合,电动机接成 Δ 型运行。按下停止按钮 SB1,电动机停止运行。

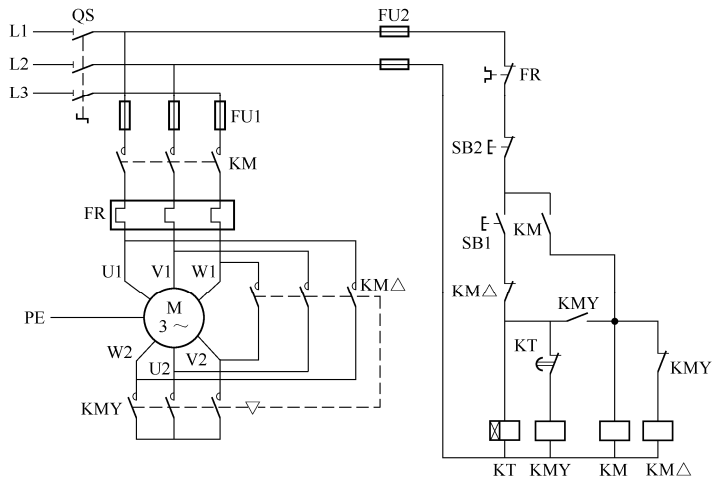


图 2-8 Y-Δ降压起动继电器-接触器控制线路

2. Y-Δ降压起动 PLC 控制的 I/O 配置（见表 2-24）与控制电路

表 2-24 PLC 的 I/O 配置

序号	类型	设备名称	信号地址	编号
1	输入	起动按钮	I0.0	SB1
2		停止按钮	I0.1	SB2
3		过载保护	I0.2	FR
4	输出	电动机接通电源接触器	Q0.0	KM1
5		星形接法	Q0.1	KM2
6		三角形接法	Q0.2	KM3

3. Y-Δ降压起动 PLC 控制的输入/输出接线（如图 2-9 所示）

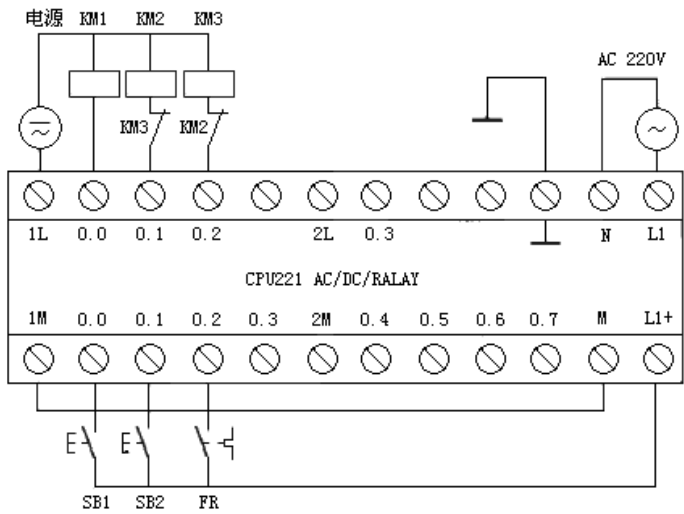


图 2-9 PLC 的 I/O 接线图

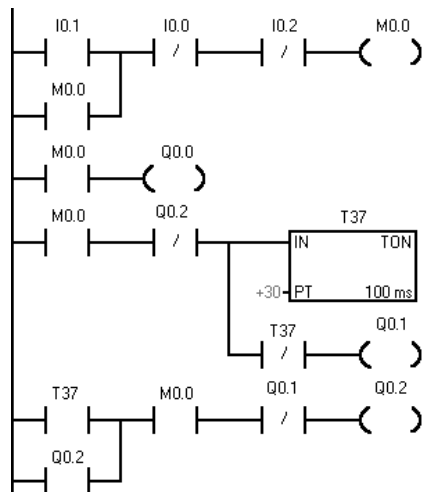
4. Y- Δ 降压起动 PLC 控制的梯形图程序设计(见图 2-10)

图 2-10 梯形图

2.3 知识拓展

PLC 程序设计常用的方法主要有经验设计法、继电器控制电路转换为梯形图法、逻辑设计法、顺序控制设计法等。

2.3.1 梯形图的经验设计法

数字量控制系统又称开关量控制系统,继电器控制系统就是典型的数字量控制系统。

可以用设计继电器电路图的方法来设计比较简单的数字量控制系统的梯形图,即在一些典型电路的基础上,根据被控对象对控制系统的具体要求,不断地修改和完善梯形图。需要多次反复地调试和修改梯形图,增加一些中间编程元件,最后才能得到一个较为满意的结果。

这种方法没有普遍的规律可循,具有很大的试探性和随意性,最后的结果不是唯一的,设计所用的时间、设计的质量与设计者的经验有很大的关系,所以有人把这种设计方法叫做经验设计法,它可以用于较简单的梯形图的设计。下面先介绍经验设计法中的一些常用的基本电路。

1. 有记忆功能的电路

起动-保持-停止电路(简称起保停电路)是典型的有记忆功能的电路,如图 2-11 所示。图中起动信号 I0.0 和停止信号 I0.1 持续为 ON 的时间一般都很短。按下起动按钮, I0.0 的常开触点接通,如果这时未按停止按钮, I0.1 的常闭触点接通, Q0.0 的线圈“通电”,它的常开触点同时接通。放开起动按钮, I0.0 的常开触点断开,“能流”经 Q0.0 的常开触点和 I0.1 的常闭触点流过 Q0.0 的线圈, Q0.0 仍为 ON,这就是所谓的“自锁”或“自保”功能。按下停止按钮, I0.1 的常闭触点断开,使 Q0.0 的线圈“断电”,其常开触点断开,以后即使放开停止按钮, I0.1 的常闭触点恢复接通状态, Q0.0 线圈仍然“断电”。这种记忆功能也可以用图 2-11 中的 S 指令和 R 指令来实现。

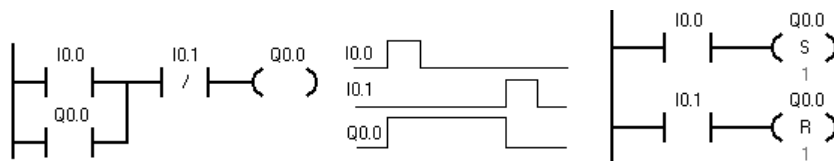


图 2-11 有记忆功能的电路

2. 定时器应用电路

【例 2-5】用定时器设计延时接通/延时断开电路，要求输入 I0.0 和输出 Q0.1 的波形如图 2-12 所示。

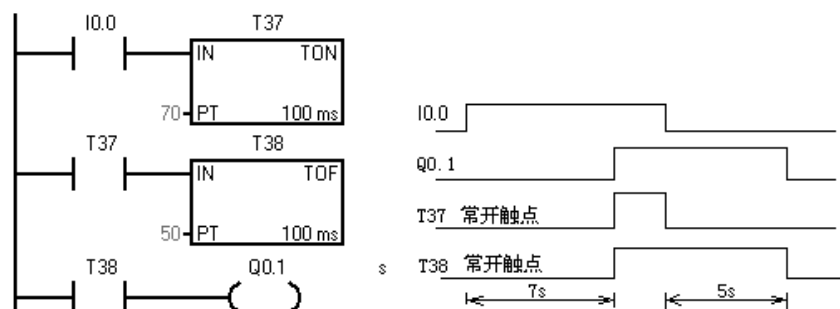


图 2-12 例 2-5 图

图中的电路用 I0.0 控制 Q0.1，I0.0 的常开触点接通后，T37 开始定时，7s 后 T37 的常开触点接通，使断开延时定时器 T38 通电，T38 的常开触点接通，使 Q0.1 的线圈通电。I0.0 变为 0 状态后 T38 开始定时，5s 后 T38 的定时时间到，其常开触点断开，使 Q0.1 变为 0 状态。

【例 2-6】用定时器设计输出脉冲的周期和占空比可调的振荡电路（即闪烁电路）。

图 2-13 中 I0.0 的常开触点接通后，T37 的 IN 输入端为 1 状态，T37 开始定时。7s 后定时时间到，T37 的常开触点接通，使 Q0.0 变为 ON，同时 T38 开始定时。5s 后 T38 的定时时间到，它的常闭触点断开，T37 因为 IN 输入电路断开而被复位。T37 的常开触点断开，使 Q0.0 变为 OFF，同时 T38 因为 IN 输入电路断开而被复位。复位后其常闭触点接通，T37 又开始定时。以后 Q0.0 的线圈将周期性地“通电”和“断电”，直到 I0.0 变为 OFF。Q0.0 的线圈“通电”和“断电”的时间分别等于 T38 和 T37 的设置值。

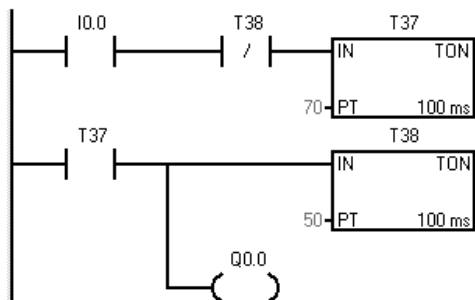


图 2-13 例 2-6 图

特殊存储器位 SM0.5 的常开触点提供周期为 1s，占空比为 0.5 的脉冲信号，可以用它来驱动需要闪烁的指示灯。

2.3.2 根据继电器电路图设计梯形图的方法

1. 基本方法

PLC 使用与继电器电路图极为相似的梯形图语言, 如果用 PLC 改造继电器控制系统, 根据继电器电路图来设计梯形图是一条捷径。这是因为原有的继电器控制系统经过长期的使用和考验, 已经被证明能完成系统要求的控制功能, 而继电器电路图又与梯形图有很多相似之处, 因此可以将继电器电路图“翻译”成梯形图, 即用 PLC 的外部硬件接线图和梯形图程序来实现继电器系统的功能。

这种设计方法一般不需要改动控制面板, 保持了系统原有的外部特性, 操作人员不用改变长期形成的操作习惯。比如本章项目实施中就是使用此法设计的。

在分析 PLC 控制系统的功能时, 可以将 PLC 想象成一个继电器控制系统中的控制箱, 其外部接线图描述了这个控制箱的外部接线, 梯形图是这个控制箱的内部“线路图”, 梯形图中的输入位 (I) 和输出位 (Q) 是这个控制箱与外部世界联系的“接口继电器”, 这样就可以用分析继电器电路图的方法来分析 PLC 控制系统。在分析时可以将梯形图中输入位的触点想象成对应的外部输入器件的触点, 将输出位的线圈想象成对应的外部负载的线圈。外部负载的线圈除受梯形图的控制外, 还可能受外部触点的控制。

将继电器电路图转换成功能相同的 PLC 的外部接线图和梯形图的步骤如下:

(1) 了解和熟悉被控设备的工艺过程和机械的动作情况, 根据继电器电路图分析和掌握控制系统的工作原理, 这样才能做到在设计和调试控制系统时心中有数。

(2) 确定 PLC 的输入信号和输出负载, 以及与它们对应的梯形图中的输入位和输出位的地址, 画出 PLC 的外部接线图。

(3) 确定与继电器电路图的中间继电器、时间继电器对应的梯形图中的位存储器 (M) 和定时器 (T) 的地址。这两步建立了继电器电路图中的元件和梯形图中编程元件的地址之间的对应关系。

(4) 根据上述对应关系画出梯形图。

2. 注意事项

梯形图和继电器电路虽然表面上看起来差不多, 实际上有本质的区别。继电器电路是全部由硬件组成的电路, 而梯形图是一种软件, 是 PLC 图形化的程序。在继电器电路图中, 由同一个继电器的多对触点控制的多个继电器的状态可能同时变化。而 PLC 的 CPU 是串行工作的, 即 CPU 同时只能处理一条与触点和线圈有关的指令。

根据继电器电路图设计 PLC 的外部接线图和梯形图时应注意以下问题:

(1) 应遵守梯形图语言中的语法规则

1) 程序应按自上而下, 从左至右的顺序编写。

2) 同一操作数的输出线圈在一个程序中不能使用两次, 不同操作数的输出线圈可以并行输出, 如图 2-14 所示。

3) 线圈不能直接与左母线相连。如果需要, 可以通过特殊内部标志位存储器 SM0.0 (该位始终为 1) 来连接, 如图 2-15 所示。

4) 适当安排编程顺序, 以减少程序的步数。

串联多的支路应尽量放在上部, 如图 2-16 所示。

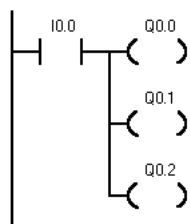


图 2-14 并行输出

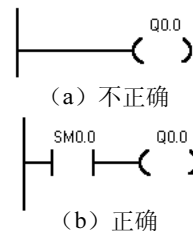
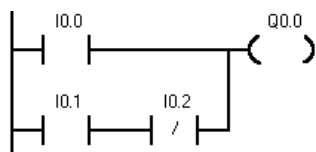
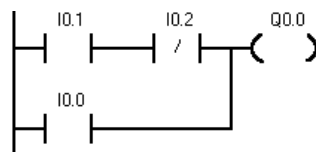


图 2-15 线圈连接



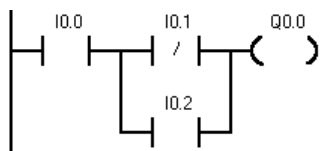
(a) 电路安排不当



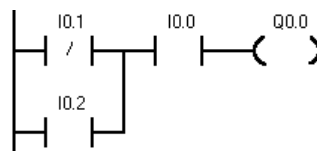
(b) 电路安排正确

图 2-16 安排编程顺序

并联多的支路应靠近左母线，如图 2-17 所示。



(a) 电路安排不当

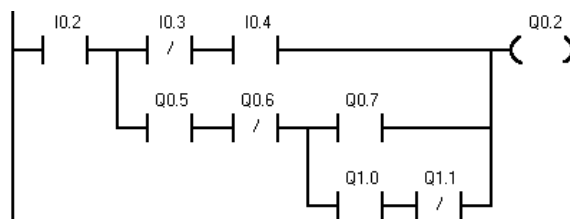


(b) 电路安排正确

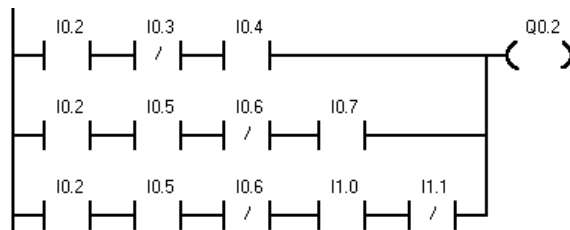
图 2-17 并联多的电路应靠近左侧母线

触点不能放在线圈的左边。

对复杂的电路，用 ALD、OLD 等指令难以编程，可重复使用一些触点画出其等效电路，然后再进行编程，如图 2-18 所示。



(a) 复杂电路



(b) 等效电路

图 2-18 复杂电路编程技巧

(2) 设置中间单元

在梯形图中,若多个线圈都受某一触点串并联电路的控制,为了简化电路,在梯形图中可以设置用该电路控制的位存储器(如图2-10中的M0.0),它类似于继电器电路的中间继电器。

(3) 尽量减少 PLC 的输入信号和输出信号

PLC 的价格与 I/O 点数有关,每一输入信号和每一输出信号分别要占用一个输入点和一个输出点,因此减少输入信号和输出信号的点数是降低硬件费用的主要措施。

与继电器电路不同,一般只需要同一输入器件的一个常开触点给 PLC 提供输入信号,在梯形图中,可以多次使用同一输入位的常开触点和常闭触点。

在继电器电路图中,如果几个输入器件的触点的串并联电路总是作为一个整体出现,可以将它们作为 PLC 的一个输入信号,只占 PLC 的一个输入点。

某些器件的触点如果在继电器电路图中只出现一次,并且与 PLC 输出端的负载串联(例如有锁存功能的热继电器的常闭触点),不必将它们作为 PLC 的输入信号,可以将它们放在 PLC 的外部的输出回路,仍与相应的外部负载串联,如图2-18中的FR过载保护。

继电器控制系统中某些相对独立且比较简单的部分,可以用继电器电路控制,这样同时减少了所需的 PLC 的输入点和输出点。

(4) 设立外部联锁电路

为了防止控制正反转的两个接触器同时动作造成三相电源短路,应在 PLC 外部设置硬件联锁电路。图2-19中的KM1与KM2、KM3与KM4的线圈不能同时通电,除了在梯形图中设置与它们对应输出位的线圈串联的常闭触点组成的联锁电路外,还在 PLC 外部设置了硬件联锁电路。

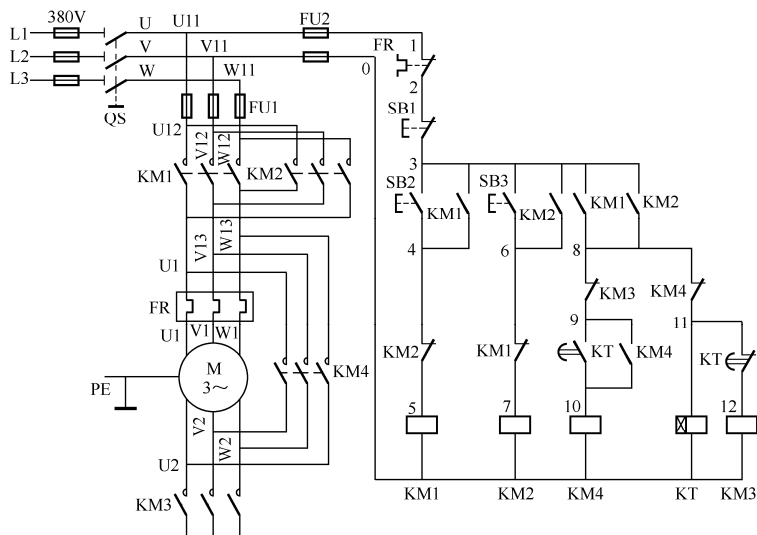


图 2-19 正反转 Y-Δ 降压起动继电-接触器控制线路

如果在继电器电路中有接触器之间的联锁电路,在 PLC 的输出回路也应采用相同的联锁电路。

(5) 梯形图的优化设计

为了减少语句表指令的指令条数,在串联电路中单个触点应放在右边(左重右轻),在并

联电路中单个触点应放在下面（上重下轻）。

（6）外部负载的额定电压

PLC 的继电器输出模块和双向晶闸管输出模块只能驱动额定电压最高 AC220V 的负载，如果系统原来的交流接触器的线圈电压为 380V，应将线圈换成 220V 的，或设置外部中间继电器。

2.3.3 电动机正反转 Y-Δ 降压起动的 PLC 控制

1. Y-Δ 降压启动继电器-接触器控制线路

有些生产机械设备，要求加工时采用正反转 Y-Δ 降压启动方式的三相鼠笼型电动机来拖动。

如图 2-19 所示。该电路能实现正、反两个方向的 Y-Δ 降压启动过程。三相鼠笼型电动机的正反转分别由 KM1 和 KM2 两个接触器来控制，KM3 和 KM4 来完成 Y-Δ 转换。

要求：电动机开始启动时接成 Y 形，延时一段时间后，自动切换到 Δ 连接运行。

2. Y-Δ 降压启动 PLC 控制的 I/O 配置（见表 2-25）与控制电路

表 2-25 PLC 的 I/O 配置

序号	类型	设备名称	信号地址	编号
1	输入	正转启动按钮	I0.0	SB1
2		反转启动按钮	I0.1	SB2
3		停止按钮	I0.2	SB3
5	输出	电动机正转接触器	Q0.0	KM1
6		电动机反转接触器	Q0.1	KM2
7		星形接法	Q0.2	KM3
8		三角形接法	Q0.3	KM4

3. Y-Δ 降压启动 PLC 控制的输入/输出接线（如图 2-20 所示）

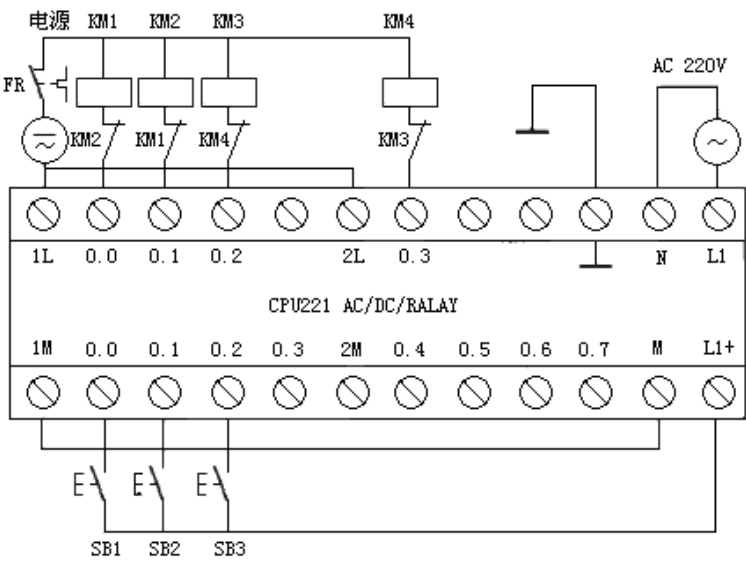


图 2-20 PLC 的 I/O 接线图

4. Y- Δ 降压起动 PLC 控制的梯形图程序设计

(1) 建立 PLC 符号表如图 2-21 所示。

符号表				
3 · 4 · 5 · 6 · 7 · 8 · 9 · 10 · 11 · 12 · 13 · 14 · 15				
	 	符号	地址	注释
1		正转启动	I0.0	正转启动
2		反转启动	I0.1	反转启动
3		停止	I0.2	停转
4		电动机正转	Q0.0	KM1
5		电动机反转	Q0.1	KM2
6		三角形接法	Q0.2	KM3
7		星形接法	Q0.3	KM4

图 2-21 符号表

(2) 梯形图如图 2-22 所示。

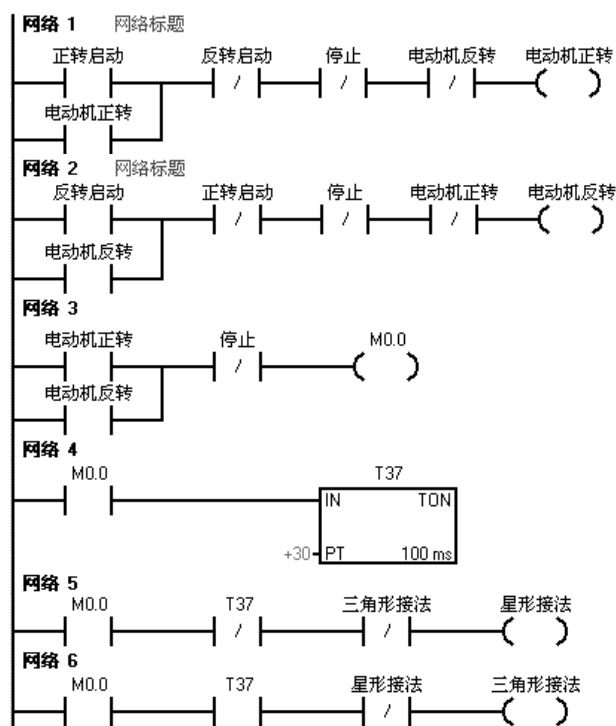


图 2-22 梯形图

2.4 技能实训

2.4.1 基本指令应用设计与调试

1. 实训目的

- (1) 掌握常用基本指令的使用方法。
- (2) 学会用基本逻辑与、或、非等指令实现基本逻辑组合电路的编程。

(3) 熟悉 S、R 指令的使用方法。

(4) 熟悉编译调试软件的使用。

2. 实训器材

PC 机一台、PLC 实训箱一台、编程电缆一根、导线若干。

3. 实训内容

西门子 S7-200 系列可编程控制器的常用基本指令有 10 条。本次实训进行常用基本指令 LD、LDN、A、AN、NOT、O、ON、ALD、OLD、= 及 S、R 指令的编程操作训练。

4. 实训步骤

(1) 实训前, 先用下载电缆将 PC 机串口与 S7-200 CPU 主机的 PORT1 端口连好, 然后对实训箱通电, 并打开 24V 电源开关。主机和 24V 电源的指示灯亮, 表示工作正常, 可进入下一步实训。

(2) 进入编译调试环境, 用指令符或梯形图输入下列练习程序。

(3) 根据程序, 进行相应的连线(接线可参见项目 1 “输入/输出端口的使用方法”)。

(4) 下载程序并运行, 观察运行结果。

练习 1 (见图 2-23):

Network 1

```
LD    I0.0
O      Q0.0
AN    I0.1
=      Q0.0
```

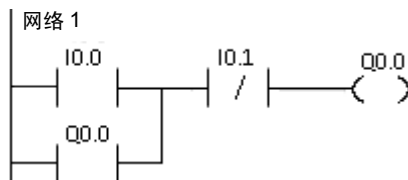


图 2-23 练习 1 图

练习 2 (见图 2-24):

Network 1

```
LD    I0.0
A      I0.1
ON     I0.2
=      Q0.0
```

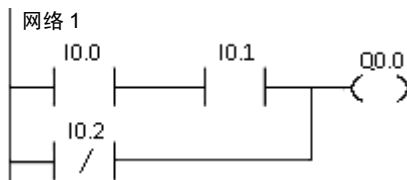


图 2-24 练习 2 图

练习 3: 在程序中要将两个程序段 (又叫电路块) 连接起来时, 需要用电路块连接指令。每个电路块都是以 LD 或 LDN 指令开始, 如图 2-25 所示。

ALD 指令:

Network 1

LD I0.0

A I0.1

LD I0.2

AN I0.3

OLD

= Q0.0

OLD 指令:

Network 1

LD I0.0

A I0.1

LDN I0.2

AN I0.3

OLD

LD I0.4

AN I0.5

OLD

= Q0.0

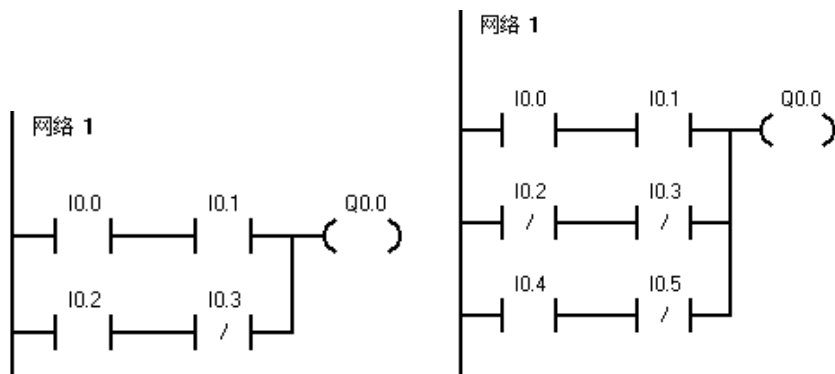


图 2-25 练习 3 图

练习 4: 置位 (S) 或复位 (R) 指令的编程, 如图 2-26 所示。

Network 1

LD I0.0

S Q0.0, 1

Network 2

LD I0.1

R Q0.0, 1

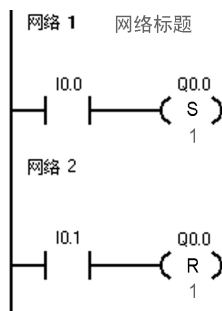


图 2-26 练习 4 图

2.4.2 定时器指令应用设计与调试

1. 实训目的

- (1) 掌握常用定时指令的使用方法。
- (2) 掌握计数器指令的使用。
- (3) 掌握定时器内部时基脉冲参数的设置。
- (4) 熟悉编译调试软件的使用。

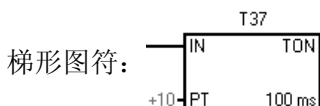
2. 实训器材

PC 机一台、PLC 实训箱一台、编程电缆一根、导线若干。

3. 实训内容

定时器指令

指令符：TONxx



数据：xx (37)：为选定的定时器号；PT (+10)：是定时器的设定值，用 4 位十进制数表示，定时单位为 0.1 秒，所以最低位是十分位。例如定时 5 秒的设定值是+50。

定时范围是 0.1~3276.7 秒。

功能：定时时间到接通定时器接点。

定时器是增 1 定时器。当输入条件为 ON 时，开始增 1 定时，每经过 0.1 秒，定时器的当前值增 1，当定时器的当前值与设定值相等时，定时时间到，定时器接点接通并保持。当输入条件为 OFF 时，不管定时器当前处于什么状态都复位，当前值恢复到 0，相应的动合接点断开。定时器相当于时间继电器。在电源掉电时，定时器复位。

4. 实训步骤

(1) 实训前，先用下载电缆将 PC 机串口与 S7-200 CPU 主机的 PORT1 端口连好，然后对实训箱通电，并打开 24V 电源开关。主机和 24V 电源的指示灯亮，表示工作正常，可进入下一步实训。

(2) 进入编译调试环境，用指令符或梯形图输入下列练习程序。

(3) 根据程序，进行相应的连线（接线可参见项目 1 “输入/输出端口的使用方法”）。

(4) 下载程序并运行，观察运行结果。

练习 1: 延时器, 如图 2-27 所示。

Network 1

LD I0.2

AN I0.3

TON T37, +30

Network 2

LD T37

= Q0.0

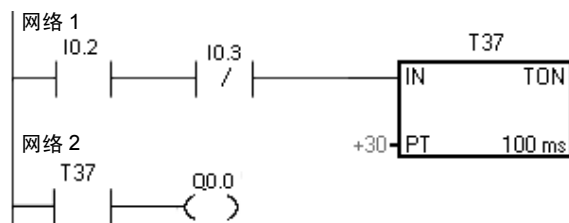


图 2-27 练习 1 图

练习 2: 秒脉冲发生器, 如图 2-28 所示。

Network 1

LDN T38

TON T37, +5

Network 2

LD T37

TON T38, +5

= Q0.0

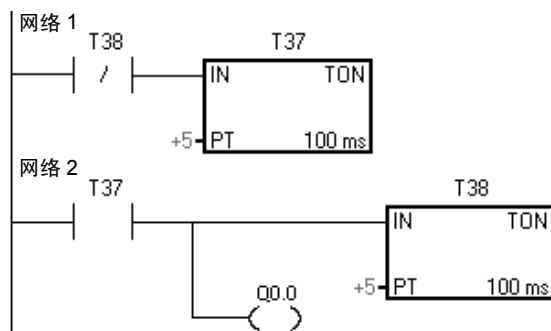


图 2-28 练习 2 图

2.4.3 电机的 Y- Δ 起动控制电路设计与调试

1. 实训目的

- (1) 掌握 PLC 功能指令的用法。
- (2) 掌握用 PLC 控制交流电机的可逆起动控制电路及 Y/ Δ 起动的电路。

2. 实训器材

PC 机一台、PLC 实训箱一台、编程电缆一根、导线若干。

3. 实训内容及步骤

设计要求

(1) 设计通过 PLC 控制电机的 Y— Δ 起动电路的程序。

Y— Δ 控制示意图如图 2-29 所示。

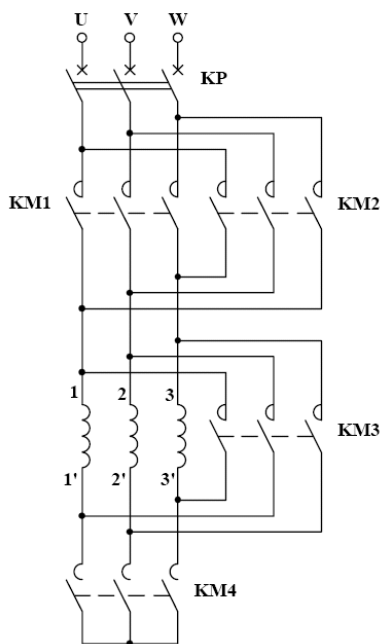


图 2-29 Y— Δ 示意图

当按下正转起动按钮时，电机正转（继电器 KM1 控制），并运行在 Y 形接法（低速运行，继电器 KM4 控制）。过 3 秒后 KM4 断开，电机运行在 Δ 接法（全速运行，继电器 KM3 控制）。

当按下停止按钮时，电机停转。

当按下反转起动按钮时，电机反转（继电器 KM2 控制），并运行在 Y 形接法（低速运行，继电器 KM4 控制）。过 3 秒后 KM4 断开，电机运行在 Δ 接法（全速运行，继电器 KM3 控制）。

(2) 确定输入、输出端口，并编写程序。

(3) 编译程序，无误后下载至 PLC 主机的存储器中，并运行程序。

(4) 调试程序，直至符合设计要求。

(5) 参考 I/O 分配接线表如表 2-26 所示。

表 2-26 I/O 分配表

输入			输出		
主机	实训模块	注释	主机	实训模块	注释
I0.0	正转	正转	Q0.0	J1 (KM1)	电机正转
I0.1	反转	反转	Q0.1	J2 (KM2)	电机反转

续表

输入			输出			
主机	实训模块	注释	主机	实训模块	注释	
I0.2	停止	停止	Q0.2	J3 (KM3)	三角形接法	
			Q0.3	J4 (KM4)	星形接法	
1M		24V	1L		0V	
				实训区	24V	24V
					0V	0V

思考与练习

1. 填空

(1) 通电延时定时器(TON)的输入(IN)_____时开始定时,当前值大于等于设定值时其定时器位变为_____,其常开触点_____,常闭触点_____。

(2) 通电延时定时器(TON)的输入(IN)电路_____时被复位,复位后其常开触点_____,常闭触点_____,当前值等于_____。

(3) 输出指令(=)不能用于_____映像寄存器。

(4) SM_____在首次扫描时为1,SM0.0一直为_____。

(5) 外部的输入电路接通时,对应的输入映像寄存器为_____状态,梯形图中对应的常开接点_____,常闭接点_____。

(6) 若梯形图中输出Q的线圈“断电”,对应的输出映像寄存器为_____状态,在输出刷新后,继电器输出模块中对应的硬件继电器的线圈_____,其常开触点_____。

2. 写出图 2-30 所示梯形图的语句表程序。

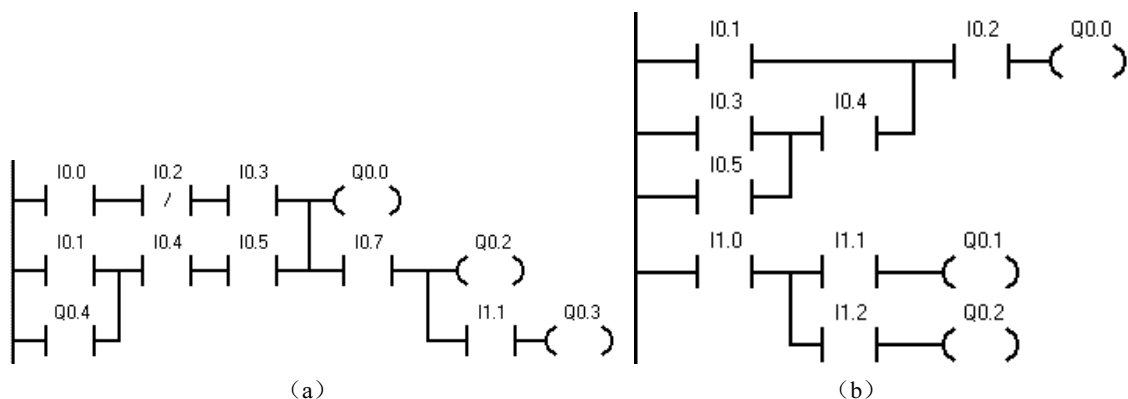


图 2-30 题 2 图

3. 根据图 2-31 画出 M0.0 的波形图。

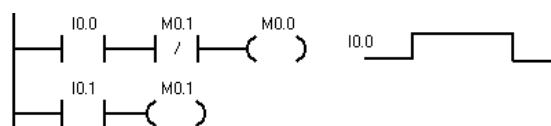


图 2-31 题 3 图

4. 根据图 2-32 画出 Q0.0 的波形图。

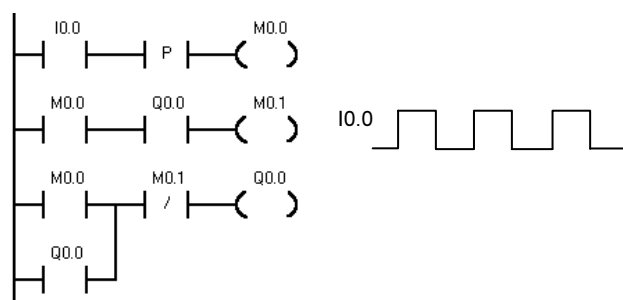


图 2-32 题 4 图

5. 设计满足图 2-33 所示时序图的梯形图。

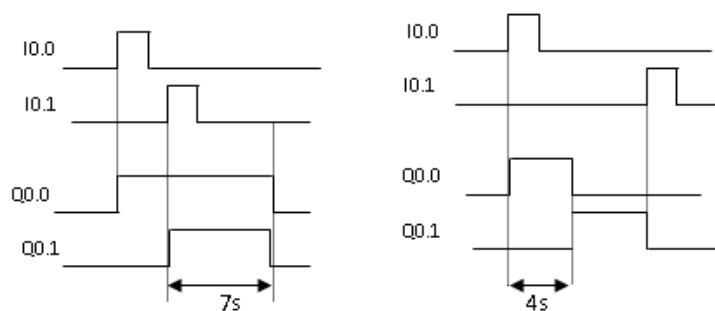


图 2-33 题 5 图

6. 画出图 2-34 所示波形对应的顺序功能图。

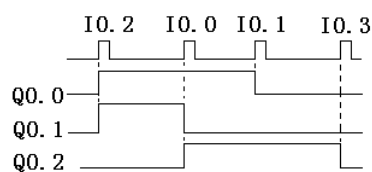


图 2-34 题 6 图