

PLC应用技术项目化教程 (西门子S7-1200)



项目4 工作台自动往返控制



★知识目标

- 1.掌握置位/复位指令、置位优先 (RS) /复位优先 (SR) 触发器及应用;
- 2.掌握边沿检测触点指令和检测信号边沿指令及应用。

★技能目标

- 1.会应用置位和复位指令编写应用程序;
- 2.学会使用监控表和强制表;
- 3.会三相异步电动机正反转控制的硬件电路接线和调试。

☆☆ 任务1 学习PLC常用位指令

※ 学习目标：

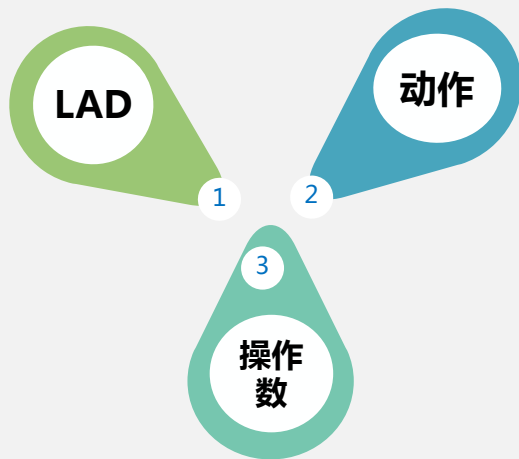
- 1.掌握置位/复位指令、置位优先（RS）/复位优先（SR）触发器及应用；
- 2.掌握边沿检测触点指令和检测信号边沿指令及应用。

☆ 任务1 学习PLC常用位指令

☆ 置位/复位指令

1. 置位指令

梯形图



将指定的地址位置位，
即变为1，并保持

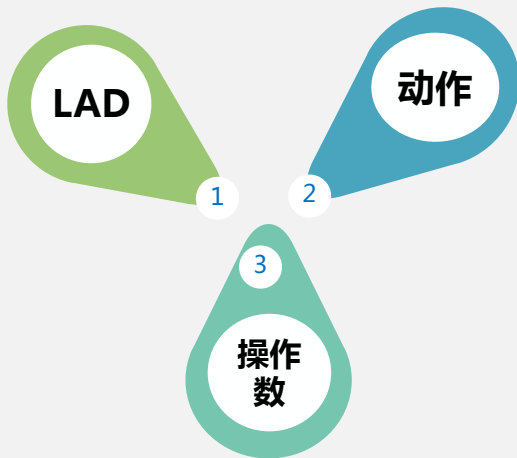
I、Q、M、D、L

☆ 任务1 学习PLC常用位指令

☆ 置位/复位指令

2. 复位指令

梯形图



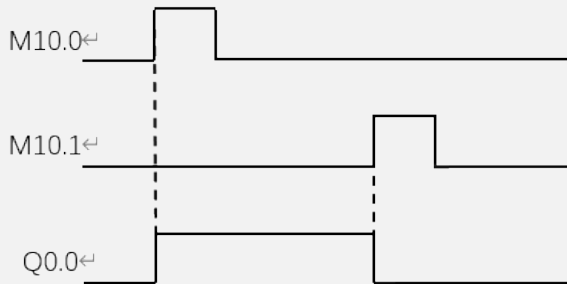
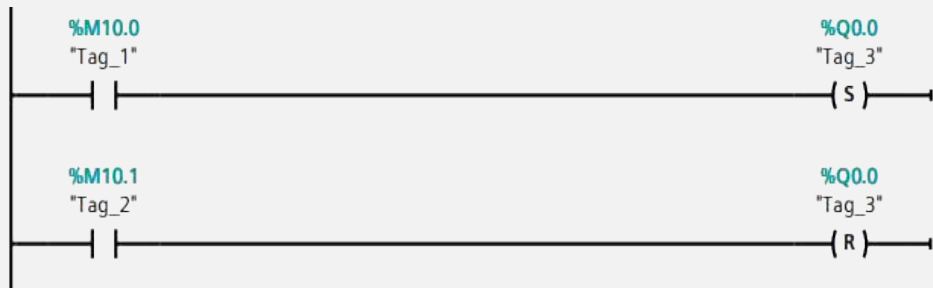
将指定的地址位复位，
即变为0，并保持

I、Q、M、D、L

☆ 任务1 学习PLC常用位指令

☆ 置位/复位指令

置位/复位指令示例图



当M10.0为1，Q0.0为1，之后，即使M10.0为0，Q0.0保持1，直到M10.1为1时，Q0.0变为0。这两条指令非常有用。置位/复位指令不一定要成对使用。

典型的梯形图之一——**置位/复位电路**

☆ 任务1 学习PLC常用位指令

“启-保-停” 电路



实现的控制完全一样

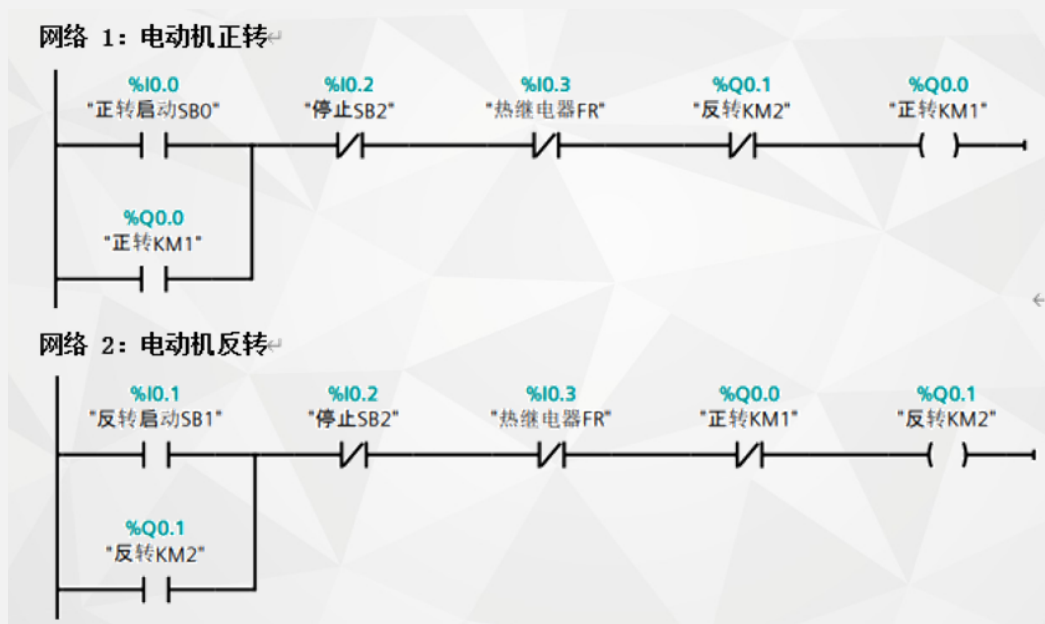
置位/复位电路



☆ 任务1 学习PLC常用位指令

思考

【例4-2】下图为实现电动机正反转的梯形图，请试用置位（S）/复位(R)指令编写出对应的梯形图程序，控制要求不变。

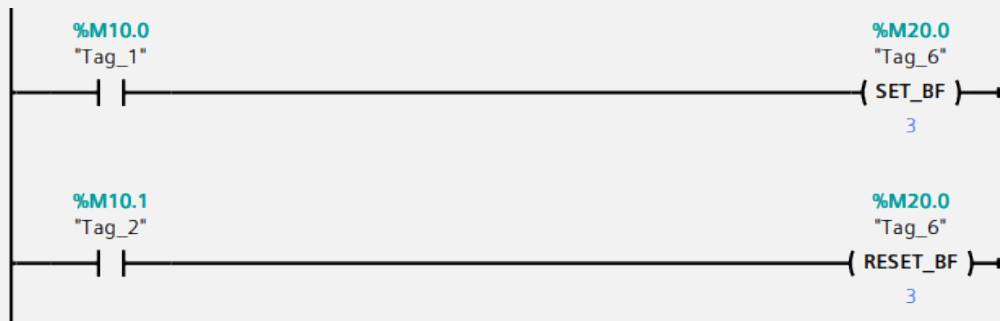


☆ 任务1 学习PLC常用位指令

☆ 多点置位 (SET_BF) / 多点复位 (RESET_BF) 指令

与置位 (S) / 复位 (R) 指令不同, 多点置位 (SET_BF) / 多点复位 (RESET_BF) 指令可以对从指定地址开始的连续多个位进行置位或复位。指令下方的数字为指定操作的位数。

如图所示, 当M10.0的常开触点闭合 (即M10.0=1) 时, 则从M20.0开始的连续3个位 (M20.0、M20.1、M20.2) 被置1并一直保持1状态。当M10.1的常开触点闭合 (即M10.1=1) 时, 将M20.0开始的连续3个位清0并一直保持0状态。



☆ 任务1 学习PLC常用位指令

☆ 置位优先 (RS) / 复位优先 (SR) 触发器

RS为置位优先触发器，它的左下角引脚是S1。SR是复位优先触发器,它的左下角引脚是R1。



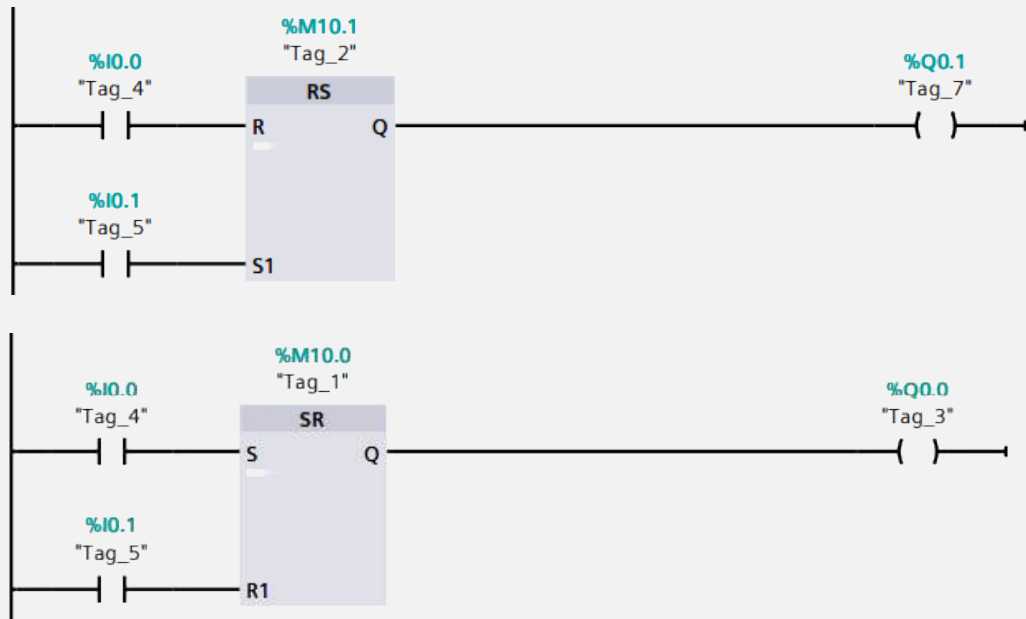
置位优先 (RS) 触发器			复位优先 (SR) 触发器		
输入		输出	输入		输出
R	S1	Q	S	R1	Q
0	0	保持之前的状态不变	0	0	保持之前的状态不变
0	1	1	0	1	0
1	0	0	1	0	1
1	1	1	1	1	0

【注意】当输入端的状态断开后，输出端状态仍然保持。如RS触发器，当R和S1为1时，输出Q为1，之后即使输入端的信号断开，输出Q端的信号仍保持1的状态。

☆ 任务1 学习PLC常用位指令

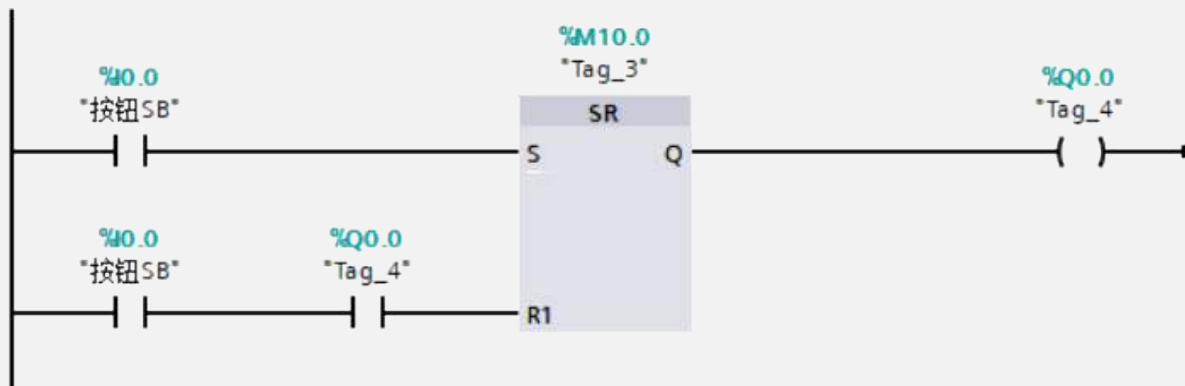
☆ 置位优先 (RS) / 复位优先 (SR) 触发器

【例4-3】 如图4-8 (a)、(b) 所示的两个梯形图中，当I0.0和I0.1常开触点同时闭合（即同时为1）时，请分别分析Q0.0和Q0.1的状态（得电还是失电状态）？



☆☆ 小试身手

【例4-4】 选用CPU1214为控制器，试编程实现用一个按钮SB控制一个指示灯L的亮和灭：第一次按下按钮灯亮，第二次按下按钮灯灭。

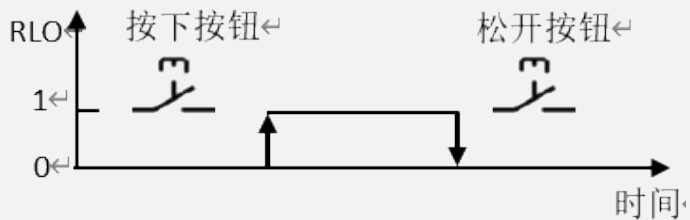


思考

编写实现该控制要求的梯形图不止以上方法，请同学们开动脑筋，你还能想出实现该控制的其他程序吗？

☆ 任务1 学习PLC常用位指令

☆ 边沿检测触点指令



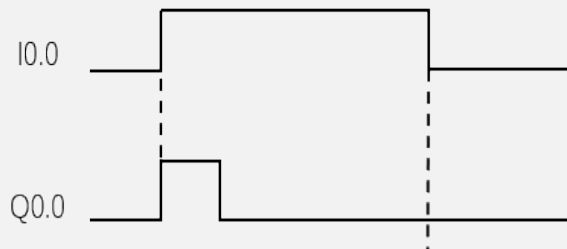
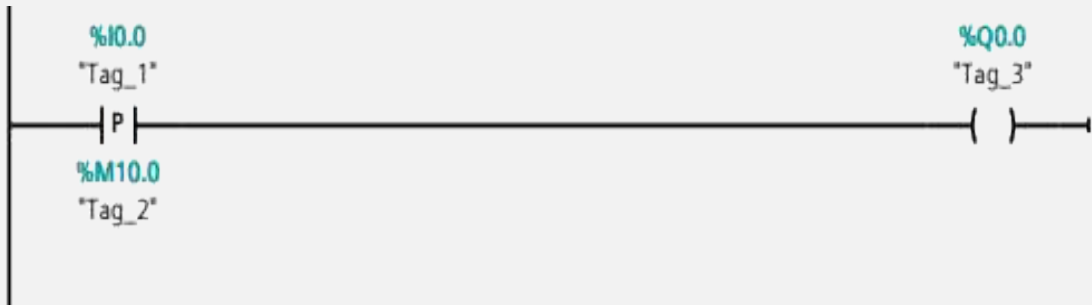
当某个信号的状态由“0”变为“1”时，则产生一个上升沿（正跳变）；当信号状态由“1”变为“0”时，则产生一个下降沿（负跳变）。例如我们按按钮，当按钮被按下时，按钮的常开触头由断开到闭合的过程就是一个“0”变为“1”的正跳变；而松开按钮，按钮的常开触头由闭合到断开的过程就是一个“1”变为“0”的负跳变；



上升沿检测触点

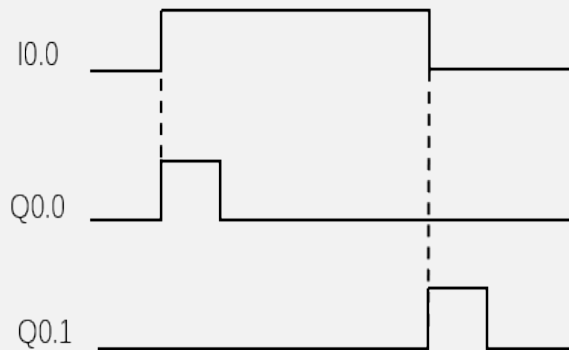
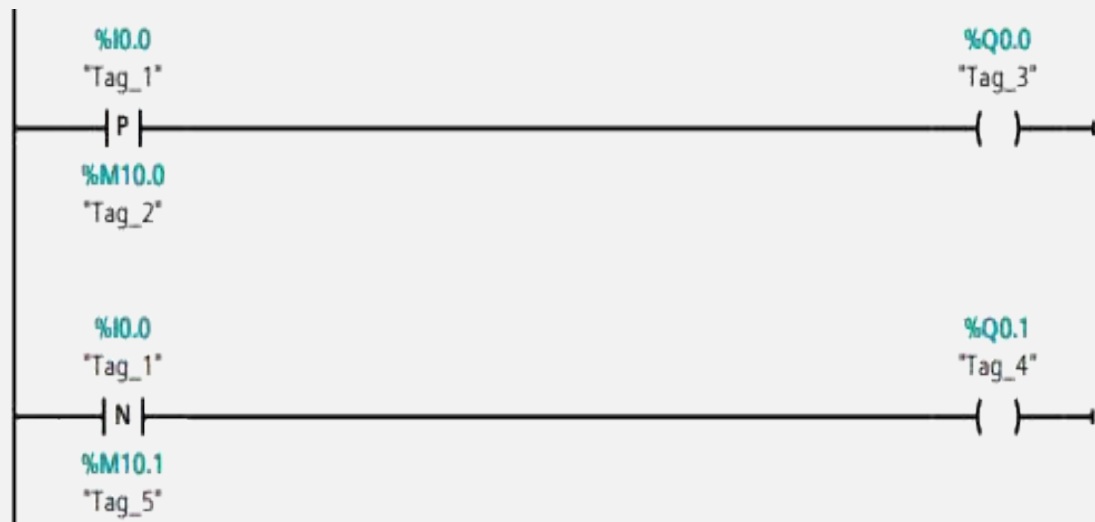


下降沿检测触点



☆ 任务1 学习PLC常用位指令

☆ 边沿检测触点指令



当按钮SB被按下时,在I0.0上产生一个上升沿,驱动Q0.0导通一个扫描周期。当松开按钮SB时,在I0.0上产生一个下降沿,此时驱动Q0.1导通一个扫描周期。

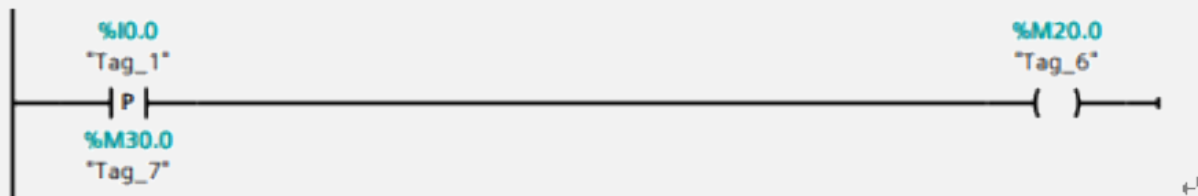
【注意】 P触点下面的M10.0和N触点下面的M10.1为边沿存储位,用来存储上一次扫描循环时I0.0的状态,这里通常用位存储器M,该地址在同一程序中不可以被重复使用。

☆ 任务1 学习PLC常用位指令

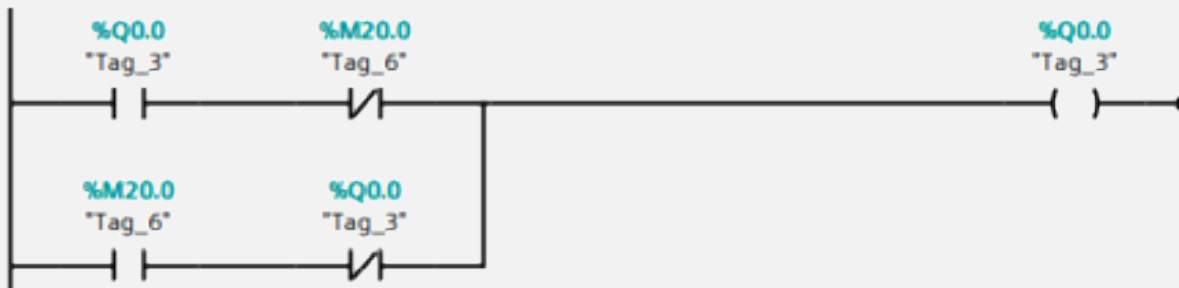
☆ 边沿检测触点指令

【例】已知I0.0外接按钮SB,Q0.0外接指示灯L,请分析图所示梯形图,说说该程序实现的功能。

程序段 1: ↵



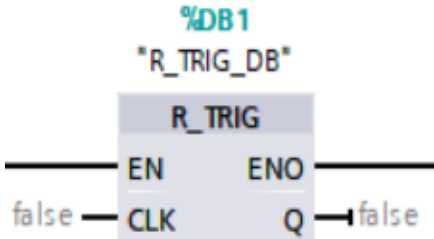
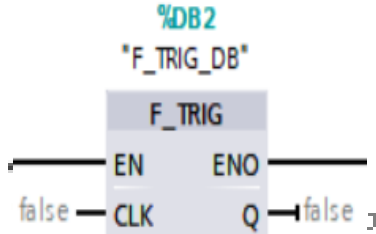
程序段 2: ↵



思考

☆ 任务1 学习PLC常用位指令

☆ 检测信号边沿指令

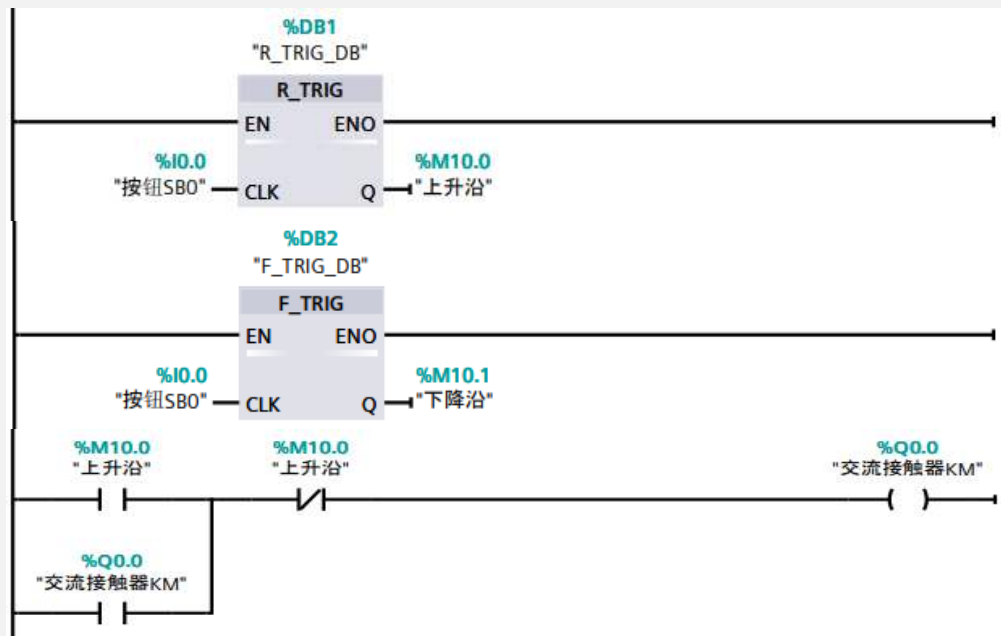
指令名称 [↗]	梯形图符号 [↗]	功能 [↗]
检测信号上升沿指令 [↗]		在信号上升沿置位变量 [↗]
检测信号下降沿指令 [↗]		在信号下降沿置位变量 [↗]

如果指令检测到输入 CLK 的状态从“0”变成了“1”（上升沿）或从“1”到“0”（下降沿），就会通过Q端输出一个扫描周期的脉冲。在其它任何情况下，该指令输出的信号状态均为“0”。

☆ 任务1 学习PLC常用位指令

☆ 检测信号边沿指令

【例】 选用CPU1214为控制器，试利用检测信号上升沿指令（R_TRIG）和检测信号下降沿指令（F_TRIG）实现电机的点动控制。控制要求：按下按钮SB0，电机转动，松开按钮SB0电机停止转动。



分析： 按钮SB0接在PLC的I0.0端子，控制电机启停的交流接触器KM接在PLC的输出端Q0.0上，控制程序梯形图如图所示。

☆ 监控表和强制表

(1) 创建监控表

当在 TIA 博图平台中的项目添加了 PLC 设备后，系统会自动为该 PLC 的 CPU 生成一个“监控和强制表”。在项目视图的项目树中打开“监控和强制表”文件夹，鼠标双击“添加新监控表”，则会自动建立一个监控表，名为“监控表 1”，鼠标双击“监控表 1”，将需要监控的变量名称输入到监控表的变量“名称”列，则其对应的地址和数据类型会自动生成。

(2) 监控表的 I/O 测试

单击“监控表 1”编辑器工具栏中“全部监视”按钮，则监控表中各变量的当前值会显示在“监视值”列中，如图 4-14 所示。同时在监控表中也可以对相关变量的值进行修改，如把 M20.0 修改为 1 的步骤：



	名称	地址	显示格式	监视值	修改值	注释	变量注释
1	"START"	%M20.0	布尔型	TRUE	TRUE		
2	"STOP"	%M20.1	布尔型	FALSE			
3	"MOTOR"	%Q0.0	布尔型	TRUE			

图 4-14 监控表监视变量值

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/858047025131006101>