

基于工业控制编程语言IEC 61131-3的 数控系统软件设计

郇极 靳阳 肖文磊 著

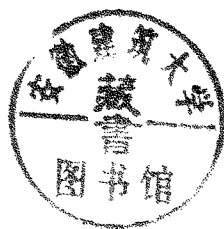
IEC 61131-3
IEC 61131-3
IEC 61131-3



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

基于工业控制编程语言 IEC 61131-3 的 数控系统软件设计

郇 极 靳 阳 肖文磊 著



北京航空航天大学出版社

内 容 简 介

IEC 61131-3 是可编程控制器(PLC)编程语言国际标准,主要用于编写工业自动化设备系统控制程序。本书介绍 IEC 61131-3 编程语言的结构和语法规则、数控系统控制原理和结构、采用 IEC 61131-3 编程语言的数控系统软件结构及其编程方法,包括:数控编程语言、译码方法、刀补计算、插补计算、坐标变换、PLC 控制、伺服现场总线、系统运行管理、人机操作界面、程序组织、模块划分、模块连接关系、控制命令传递、模块信息交换、数据结构等。此外,本书还提供了大量编程示例。

本书可作为工业自动化和计算机控制专业研究生教材或教学参考书,亦可作为数控系统开发人员的专业工具书。

图书在版编目(CIP)数据

基于工业控制编程语言 IEC 61131-3 的数控系统软件设计 / 郇极, 靳阳, 肖文磊著. --北京:北京航空航天大学出版社, 2011. 8

ISBN 978-7-5124-0485-4

I. ①基… II. ①郇… ②靳… ③肖… III. ①工业控制系统:数字控制系统—系统软件—软件设计 IV.

①TB4 ②TP273

中国版本图书馆 CIP 数据核字(2011)第 122507 号

版权所有,侵权必究。

基于工业控制编程语言 IEC 61131-3 的数控系统软件设计

郇 极 靳 阳 肖 文 磊 著

责任编辑 李文轶

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱: bhpess@263.net 邮购电话:(010)82316936

涿州市新华印刷有限公司印装 各地书店经销

*

开本:787 mm×1 092 mm 1/16 印张:12.5 字数:320 千字

2011 年 8 月第 1 版 2011 年 8 月第 1 次印刷 印数:4 000 册

ISBN 978-7-5124-0485-4 定价:28.00 元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

前 言

IEC 61131-3 是国际电工委员会(IEC)颁布的可编程控制器(PLC)编程语言国际标准,用于规范可编程控制器编程工具和应用控制程序开发。在 PLC 技术的发展过程中,形成了多种编程语言,其中 5 种编程语言获得广泛应用:指令表(IL, Instruction List)、结构化文本(ST, Structured Text)、梯形图(LD, Ladder Diagram)、功能块图(FBD, Function Block Diagram)和顺序功能图(SFC, Sequential Function Chart)。IEC 61131-3 标准以这 5 种编程语言为基础,对其变量类型定义、语法规则、程序组织结构等做出了统一规定。同时将计算机技术领域的先进编程技术引入到可编程控制器编程,形成工业控制编程语言标准。大大增强了可编程控制器编程语言的功能、结构化和模块化水平,以及提高了编程效率和程序的可重用性。

IEC 61131-3 标准已经被工业自动化控制技术领域广泛采用,国外有一批 PLC 编程工具供应商专门提供 IEC 61131-3 编程语言工具。PLC 控制产品生产厂商通常购买这些编程工具的使用权,应用于自己的产品中。PLC 控制器用户采用 IEC 61131-3 语言编写自己的应用程序,完成工程开发。

本书作者长期从事数控系统软件编程技术研究和控制程序开发工作,一直关注 IEC 61131-3 标准编程语言的发展,探索使用 IEC 61131-3 编程语言编写数控系统软件的方法,并开展相关的研究和试验工作,开发出基于 IEC 61131-3 编程标准的数控系统样机。实际开发成果表明,IEC 61131-3 编程语言为数控系统软件开发提供了一种新的编程方法和工具。它比使用常规计算机编程语言编写数控系统软件具有更大的优越性。

IEC 61131-3 为数控系统软件提供了一种模块化和可视化的编程语言。使用 IEC 61131-3 的功能块图编程方法,可以将控制功能划分和封闭在功能块内,通过清晰的连接关系构成系统,同时也方便模块的更换、增加和去除,使数控系统软件成为可重构的开放式系统。功能块图对内部和外部数据变量的操作有严格的权限规定,变量具有读写属性和作用域,自动保证了模块功能的封闭性和数据的安全性。本书介绍 IEC 61131-3 编程语言的结构和语法规则、数控系统的软件结构、采用 IEC 61131-3 的数控系统软件结构、编程方法和程序示例。

本书的章节安排如下:

第 1 章为概述,简要介绍 IEC 61131-3 标准的产生和组成结构、数控系统软件的控制原理和结构、IEC 61131-3 作为数控系统软件开发语言的技术优越性。

第 2 章介绍了 IEC 61131-3 标准编程语言的 5 种编程方式及其程序示例。

第 3 章介绍了数控系统及其控制软件的整体构架。

第 4 章给出了 IEC 61131-3 编程语言的标准体系、程序结构、变量类型。

第 5 章详细叙述了基于功能块图和结构化文本语言的数控系统软件设计。

第 6 章介绍了数控系统软件的控制原理、实现方法、程序和数据结构的设计流程。

第 7 章介绍了数控系统数据和参数的结构。

附录 A 介绍了 ISO 6983 数控编程指令国际标准。

附录 B 介绍了本书使用的自定义 G 指令代码。

本书既是一本介绍使用 IEC 61131-3 标准编写数控系统软件的书，同时也是一本学习数控系统控制软件的原理和编程方法的书。使用 IEC 61131-3 图形编程语言和数据关系描述，使我们更容易理解和掌握数控系统软件的结构，特别是功能模块的划分、接口、数据流、以及复杂的实时数据处理关系。读者通过本书学习，掌握了数控系统软件的原理和结构以后，对使用其他编程语言(例如 C 语言)编写数控系统软件也会有很大帮助。本书所介绍的数控机床控制系统模块化软件结构和设计方法也适用于基于数字传动技术的数控机械控制系统设计，例如数控印刷机、纺织机等。

郇 极

2011 年 5 月于北航

目 录

第 1 章 概 述	1
1.1 IEC 61131-3 编程语言标准.....	1
1.2 数控系统和控制软件.....	2
1.3 本书撰写特点.....	3
第 2 章 IEC 61131-3 标准编程语言简介	4
2.1 指令表(IL).....	5
2.2 结构化文本(ST).....	7
2.3 梯形图(LD).....	8
2.4 功能块图(FBD).....	9
2.5 顺序功能图(SFC).....	13
第 3 章 数控系统和软件结构	16
3.1 数控机床和控制系统.....	16
3.2 数控系统软件结构.....	16
3.2.1 控制数据流.....	18
3.2.2 操作和运行控制.....	18
3.2.3 数控系统软件的功能图描述.....	18
第 4 章 IEC 61131-3 标准体系和语言结构	21
4.1 软件体系结构.....	21
4.1.1 软件模型.....	21
4.1.2 程序结构.....	23
4.1.3 程序组织单元(POU).....	24
4.2 数据类型和变量.....	24
4.2.1 数据类型.....	24
4.2.2 变 量.....	26

第 5 章 基于功能块图和结构化文本语言的数控系统软件设计	30
5.1 功能块图 (FBD) 语言.....	30
5.1.1 功能块的定义和变量声明.....	31
5.1.2 程序示例 1.....	33
5.1.3 程序示例 2.....	34
5.1.4 程序示例 3.....	36
5.2 结构化文本 (ST) 语言.....	38
5.2.1 程序语句.....	38
5.2.2 标准功能.....	40
5.2.3 典型语句示例.....	42
5.3 数控系统软件模块和连接.....	45
5.3.1 功能块.....	45
5.3.2 数据电缆.....	45
5.3.3 组件和组件数据.....	46
5.3.4 数据标记.....	47
第 6 章 数控系统软件设计	49
6.1 系统总体结构.....	49
6.1.1 数控加工程序预处理功能库.....	51
6.1.2 运动和 PLC 控制程序.....	51
6.1.3 操作和系统运行管理任务.....	51
6.2 系统数据结构.....	54
6.2.1 常数全局变量.....	54
6.2.2 系统全局变量.....	54
6.2.3 参 数.....	55
6.2.4 数据电缆.....	55
6.2.5 组件变量.....	55
6.2.6 功能库变量.....	55
6.3 数控加工程序预处理功能库.....	56
6.3.1 数控加工程序和指令.....	56
6.3.2 数控加工程序读入模块.....	59

6.3.3	译码器	62
6.3.4	编程坐标系处理	66
6.3.5	刀具半径和长度补偿	72
6.3.6	写控制指令缓冲区 FIFO	75
6.4	数控加工程序预处理功能库的运行控制	77
6.4.1	数控加工程序预处理功能库的调用模块	77
6.4.2	数控加工程序预处理功能库的调用时序和控制	78
6.4.3	程序示例	79
6.5	运动和 PLC 控制	82
6.5.1	读控制指令缓冲区 FIFO	82
6.5.2	插补器组件	83
6.5.3	手动进给	115
6.5.4	坐标变换模块	121
6.5.5	机床误差补偿	128
6.5.6	机床传动匹配	132
6.5.7	现场总线驱动	136
6.5.8	伺服状态监视	141
6.5.9	PLC 控制	143
6.6	操作与运行管理	146
6.6.1	操作和显示 (HMI)	146
6.6.2	系统运行管理	154
第 7 章	系统数据定义	159
7.1	常数全局变量	160
7.2	系统全局变量	163
7.3	参 数	168
7.3.1	配置参数	168
7.3.2	系统参数	169
7.3.3	刀具参数	170
7.3.4	坐标系参数	170
7.4	数据电缆	171

7.4.1	主程序数据电缆定义	172
7.4.2	系统全局数据电缆定义	176
7.4.3	数控加工程序预处理功能库数据电缆	179
7.5	主程序和功能库程序内部变量数据结构	182
7.5.1	主程序组件变量数据结构	182
7.5.2	数控加工程序预处理功能库内部变量数据结构	184
附录 A	ISO 6983 数控编程指令标准	186
A.1	字符集	186
A.2	G 指令集	188
A.3	M 指令集	190
附录 B	自定义指令代码	191
参考文献		192

第 1 章 概 述

20 世纪 80 年代是可编程控制器(PLC)的一个快速发展阶段,在应用过程中形成了多种编程语言。为了促进和支持可编程控制器的发展和应用,国际电工委员会(IEC, International Electro-Technical Commision)开始着手制定可编程控制器国际标准,经过多年工作,于 1992 年开始陆续发布标准,编号为 IEC 1131。编程语言标准是其中的一个主要组成部分,于 1993 年颁布,编号为 IEC 1131-3。后来由于 IEC 标准编号系统的扩展,标准编号变更为 IEC 61131,其编程语言标准编号变更为 IEC 61131-3。

IEC 61131 标准由 8 个部分组成,其作用和发布情况如下:

- IEC 61131-1 通用信息(1992);
- IEC 61131-2 设备特性(1992);
- IEC 61131-3 编程语言(1993);
- IEC 61131-4 用户导则(1995);
- IEC 61131-5 通信(2000);
- IEC 61131-6 现场总线通信(现场总线标准,尚待完成);
- IEC 61131-7 模糊控制编程(2000);
- IEC 61131-8 编程语言和实现导则(2001)。

中国根据国际标准体系等效原则,颁布了对应的国家标准 GB/T 15969.1~8,等效于 IEC 61131-1~8。

1.1 IEC 61131-3 编程语言标准

在 20 世纪 80 年代 PLC 技术的迅速发展过程中,形成了多种编程语言,其中 5 种编程语言获得了广泛应用。IEC 61131-3 标准以这 5 种编程语言为基础,对其数据类型、变量结构、语法规则、程序组织等做出了统一规定,形成工业控制编程语言标准。IEC 61131-3 标准将计算机技术领域的先进编程技术引入到可编程控制器,例如:结构化和模块化编程技术、面向对象编程技术、可视化编程技术等。同时将这些技术与传统的可编程控制器编程技术结合,例如支持梯形图和指令表编程语言。极大地增强了编程语言的功能,提高了编程效率、质量和可读性。它注重工业控制语言设计的模块化结构、具有很强的兼容性、互换性、开放性、重复使用特性,为基于可编程控制器的工业控制装置和系统控制软件设计提供了新的方法。并且为设备的制造、安装和维护带来极大方便。

IEC 61131-3 标准规定 5 种编程语言:指令表(IL, Instruction List)、结构化文本(ST, Structured Text)、梯形图(LD, Ladder Diagram)、功能块图(FBD, Function Block Diagram)和顺序功能图(SFC, Sequential Function Chart)。国外有一批专业的 PLC 编程工具供应商提

供 IEC 61131-3 编程语言工具, 例如: 德国 3S 公司的 CoDeSys、德国 KW 公司的 MULTIPROG、德国 Infoteam 公司的 OpenPCS 等。PLC 控制产品的生产商通常购买这些编程工具的使用权, 将其应用在自己的产品中。本书的示例程序就是在德国 Beckhoff 自动化系统公司的可编程控制系统 TwinCAT 上编写的。TwinCAT 使用了德国 3S 公司的 IEC 61131-3 标准编程工具 CoDeSys。

1.2 数控系统和控制软件

数控机床由计算机控制产生工作台和主轴运动, 执行加工程序规定的运动顺序、速度和轨迹, 完成零件的加工。数控机床由 3 个主要部分组成: 机床机械本体、伺服驱动(伺服装置和电机)、数控系统(装置)。数控系统也称 CNC 控制系统(Computer Numerical Controller), 是数控机床的核心控制装置, 它是一个专用的控制计算机或者是在 PC 计算机结构基础上构建的控制计算机。数控系统通过控制伺服装置和电机, 产生机床工作台和主轴的运动。数控系统的主要部件与计算机相同, 由 CPU、内部存储器、外部存储器、显示器、键盘、外部设备接口组成。

为了实现数控机床的控制, 必须为它设计开发专门的控制程序, 称为数控系统控制软件。控制软件的主要任务包括: 实时操作系统或中断调度系统、编写和管理数控加工程序、机床工作台运动轨迹的控制计算、主轴控制、辅助设备控制(冷却泵、夹具等)、人机操作界面等。数控系统软件使用通用的计算机编程语言编写, 目前主要采用 C 语言编写。数控系统的性能取决于计算机硬件和数控软件功能, 而数控软件的功能、可靠性和开发成本则取决于它的编程语言。

数控系统编程语言与计算机编程语言同步发展, 在历史上经历以下 3 个主要阶段:

- 20 世纪 70 年代及之前: 机器代码;
- 20 世纪 70 年代中期—80 年代中期: 汇编语言;
- 20 世纪 80 年代中期—目前: C 语言。

IEC 61131-3 是用于工业控制系统的编程语言国际标准, 它为工业控制系统软件开发提供了设计工具和设计规范。代表了工业控制软件设计技术的进步和发展方向。也是未来数控系统软件编程的发展方向。

本书作者长期从事数控系统软件编程技术研究和控制程序开发工作, 关注 IEC 61131-3 标准编程语言的发展, 探索使用 IEC 61131-3 编程语言编写数控系统软件的方法, 并开展相关的研究和试验工作, 开发出基于 IEC 61131-3 编程标准的数控系统样机。实际开发成果表明, IEC 61131-3 编程语言为数控系统软件开发提供了一种新的语言和编程方法。使用 IEC 61131-3 语言编写数控系统软件的主要优越性是:

- ① 具有面向工业自动化设备控制的图形化编程功能;
- ② 具有计算机高级编程语言的结构化和面向对象特征, 提供强大的系统规划、功能模块划分、组合和封装功能;
- ③ 具有处理复杂和高精度计算的能力;
- ④ 具有丰富的程序循环控制和逻辑运算处理功能;

⑤ 具有标准化的系统组件数据通信功能;

⑥ 具有标准化的外部设备接口功能。

IEC 61131-3 的功能块图编程方法提供图形编程功能,用图形表示功能块的输入、输出以及与其他模块的连接关系,可以直接形成系统的结构图。使用 IEC 61131-3 的功能块图编程方法,可将控制功能划分和封闭在功能块内,通过清晰的连接关系构成系统,同时也便于模块的更换、增加和去除,使数控系统软件成为可重构的开放式系统。功能块图对内部和外部数据变量的操作有严格的权限规定,变量具有读写属性和作用域,自动保证了模块功能的封闭性和数据的安全性。

功能块内部可以嵌套子功能块,也可以在功能块内部采用结构化文本语言(ST)编程,实现强大的数学运算和逻辑处理功能。当需要扩充系统功能时,可以定义和开发一个新的功能块,它具有与系统其他相关模块连接的输入、输出接口,经过测试后,可以将其连接到系统中,成为数控系统软件的组成部分。也可以在不改变模块接口定义的情况下,开发具有新功能的模块,替换原有的模块,集成到系统中。以上编程特征特别适合将机床用户要求的特殊功能集成到标准数控系统中,这部分的编程工作可以由用户完成。使用 IEC 61131-3 编程语言比使用常规计算机编程语言编写数控系统软件具有更大的优越性。

1.3 本书撰写特点

本书注重介绍数字控制系统的控制原理、系统结构、程序组织、模块划分、模块连接关系、控制命令传递、模块工作状态传递、数据组织等。为了便于读者理解以及由于篇幅限制,省略一些具体的计算、逻辑处理、错误报警、命令和状态互锁等程序细节。它不影响读者对 IEC 61131-3 编程语言编程方法的了解。

作者采用功能块图(FBD)和结构化文本(ST)混合编程技术完成数控系统软件开发。利用 IEC 61131-3 标准的功能特点,针对数控软件设计,提出了“数据电缆”和“数据标记”变量类型。它也是用 IEC 61131-3 编程语言编写数控系统软件的关键技术之一。

本书内容也非常适合数控系统控制软件的设计原理和编程方法的学习。使用 IEC 61131-3 的图形编程语言和数据关系描述,使我们更容易理解数控系统软件结构,特别是功能模块的划分、接口、数据流以及复杂的实时数据处理关系。读者通过本书学习,掌握了数控系统软件的原理和结构以后,也可以使用其他编程语言编写数控系统软件,例如 C 语言。

第 2 章 IEC 61131—3 标准编程语言简介

IEC 61131—3 编程标准规定了 5 种编程语言。其中包括 2 种文本编程语言：指令表(IL, Instruction List)、结构化文本(ST, Structured Text)和 3 种图形编程语言：梯形图(LD, Ladder)、功能块图(FBD, Function Diagram Block)、顺序功能图(SFC, Sequential Function Chart)。

本章以一个典型的继电器控制电路为例，介绍 IEC 61131—3 标准的 5 种编程语言。图 2.1(a)是一个用于控制指示灯开关的继电器控制电路，组成如下：

- HL 为指示灯；
- KM 为继电器；
- SB0 为点亮指示灯控制按钮；
- SB1 为熄灭指示灯控制按钮。

该继电器控制电路的控制动作如下：

(1) 按下点亮指示灯控制按钮 SB0，继电器线圈 KM 通电，其中一对触点接通指示灯 HL，指示灯亮，另外一对触点并联 SB0，产生电路自锁；

(2) 按下熄灭指示灯控制按钮 SB1，继电器线圈 KM 断电，它的触点断开指示灯 HL，指示灯熄灭。

完成指示灯开关控制的 PLC 控制单元如图 2.1(b)所示，点亮指示灯控制按钮 SB0 连接 PLC 的输入端口 IX0.0，熄灭指示灯控制按钮 SB1 连接输入端口 IX0.1，输出端口 QX0.0 连接指示灯 HL。以下分别介绍使用 IEC 61131—3 标准的 5 种编程语言完成指示灯开关控制的编程方法。

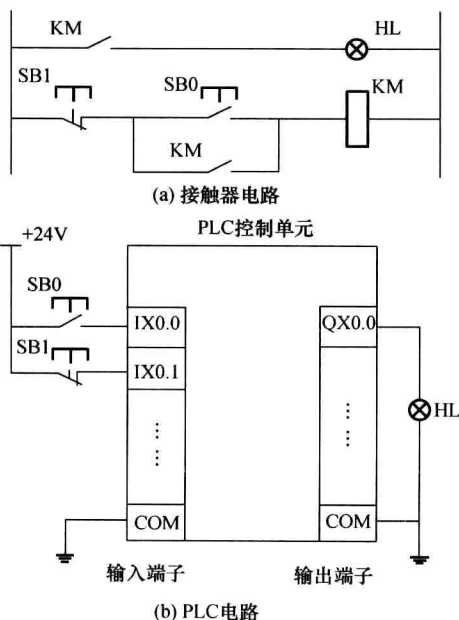


图 2.1 接触器和 PLC 控制电路

2.1 指令表(IL)

采用指令表编程语言实现本例指示灯开关控制的示例程序名称为 `il_onoff`，由如下 2 部分组成。

(1) 变量声明

```
PROGRAM il_onoff
VAR
    SB0 AT %IX0.0:BOOL;
    SB1 AT %IX0.1:BOOL:=TRUE;
    km:BOOL;
    HL AT %QX0.0:BOOL;
END_VAR
```

(2) 程序体

```
LD    SB0
OR    km
AND   SB1
ST    km
ST    HL
```

在变量声明部分，定义了 4 个变量：

- **SB0**：输入变量，连接输入端口 `IX0.0`，变量类型为 `BOOL`；
- **SB1**：输入变量，连接输入端口 `IX0.1`，变量类型为 `BOOL`，初始化为 `TRUE`；
- **km**：中间变量，变量类型为 `BOOL`；
- **HL**：输出变量，连接输出端口 `QX0.0`，变量类型为 `BOOL`。

指令表编程语言起源于计算机编程汇编语言。每行文字表示一条控制指令，控制指令由操作符和变量组成。PLC 控制系统运行时由上自下逐行执行控制程序，完成控制动作。本例中程序语句功能如下：

- **LD SB0**：将输入端口 `%IX0.0` 的状态值写入运算累加器；
- **OR km**：运算累加器与中间变量 `km` 做“或”逻辑操作；
- **AND SB1**：将读入输入端口 `%IX0.1` 的状态值与当前运算累加器内容做“与”逻辑操作；
- **ST km**：将当前累加器状态写到中间变量；
- **ST HL**：将当前累加器状态写到输出端口 `%QX0.0`。

PLC 控制器按照设定的控制周期循环执行以上控制语句，实现要求的控制动作。

指令表编程语言主要包括如下控制指令。

(1) 逻辑运算操作符

表 2.1 是逻辑运算操作符的功能说明和示例。

表 2.1 逻辑运算操作符

操作符	语 义	示 例
LD	将操作数装入到运算累加器 CR	LD SB0
ST	将 CR 内的值赋值给操作数	ST HL
S	若 CR=1, 则将操作数置位为 1	S SB0
R	若 CR=1, 则将操作数复位为 0	R SB0
AND	逻辑与	AND SB0
OR	逻辑或	OR SB0
XOR	逻辑异或	XOR SB0
NOT	逻辑取反	NOT SB0

(2) 数学运算操作符

表 2.2 是数学运算操作符的功能说明和示例。

表 2.2 数学运算操作符

操作符	语 义	示 例
ADD	CR 内的值加操作数, 并将结果存入 CR	ADD Var1
SUB	CR 内的值减操作数, 并将结果存入 CR	SUB Var1
MUL	CR 内的值乘以操作数, 并将结果存入 CR	MUL Var1
DIV	CR 内的值除以操作数, 并将结果存入 CR	DIV Var1

(3) 判断和跳转操作符

表 2.3 是判断和跳转操作符的功能说明和示例。

表 2.3 判断和跳转操作符

操作符	语 义	示 例
GT	如果 CR 内的值 > 操作数, 则将 CR 内的值设为 TRUE(大于)	GT Var1
GE	如果 CR 内的值 \geq 操作数, 则将 CR 内的值设为 TRUE(大于等于)	GE Var1
EQ	如果 CR 内的值 = 操作数, 则将 CR 内的值设为 TRUE(等于)	EQ Var1
NE	如果 CR 内的值 \neq 操作数, 则将 CR 内的值设为 TRUE(不等于)	NE Var1
LE	如果 CR 内的值 \leq 操作数, 则将 CR 内的值设为 TRUE(小于等于)	LE Var1
LT	如果 CR 内的值 < 操作数, 则将 CR 内的值设为 TRUE(小于)	LT Var1
JMP	跳转	JMP A (A 为语句标识符)

(4) 子程序调用和返回操作符

表 2.4 是子程序调用和返回操作符的功能说明和示例。

表 2.4 子程序调用和返回操作符

操作符	语义	示例
CAL	调用功能、功能块或子程序	CAL PROG1 ()
RET	从被调用的功能、功能块或子程序返回	RET

2.2 结构化文本(ST)

采用结构化文本编程语言实现本例指示灯开关控制的示例程序名称为 `st_onoff`，由如下 2 部分组成：

(1) 变量声明

```
PROGRAM st_onoff
VAR
    SB0 AT %IX0.0:BOOL;
    SB1 AT %IX0.1:BOOL:=TRUE;
    km:BOOL;
    HL AT %QX0.0:BOOL;
END_VAR
```

(2) 程序体

```
km:=(SB0 OR km) AND SB1;
HL:=km;
```

变量声明部分的定义与“2.1 指令表”中的变量定义相同。结构化文本编程语言起源于计算机高级语言，非常类似于 PASCAL 语言。采用面向变量和操作的文本语言编程，程序结构非常清晰，具有非常丰富的数学运算、逻辑处理功能和程序循环控制功能。PLC 控制系统运行时由上自下逐条执行程序语句，完成控制动作。PLC 控制器按照设定的控制周期循环执行控制语句。本例程序语句的功能如下：

- `km:=(SB0 OR km) AND SB1`：读入操作按钮 SB0 和 SB1 状态(输入端口%IX0.0 和 %IX0.1)，并与中间变量 km 做逻辑运算，然后将运算结果赋值给 km；
- `HL:=km`：将 km 状态赋值给输出变量(输出端口%QX0.0)。

结构化文本语言是本书介绍数控系统软件编程技术所使用的编程语言，将在“5.2 结构化文本(ST)语言”部分作详细介绍。

2.3 梯形图 (LD)

采用梯形图编程语言实现本例指示灯开关控制的示例程序名称为 ld_onoff, 由如下 2 部分组成:

(1) 变量声明

```
PROGRAM ld_onoff
VAR
    SB0 AT %IX0.0:BOOL;
    SB1 AT %IX0.1:BOOL:=TRUE;
    km:BOOL;
    HL AT %QX0.0:BOOL;
END_VAR
```

(2) 梯形图

图 2.2 是采用梯形图编程语言编写的本例指示灯开关控制程序。

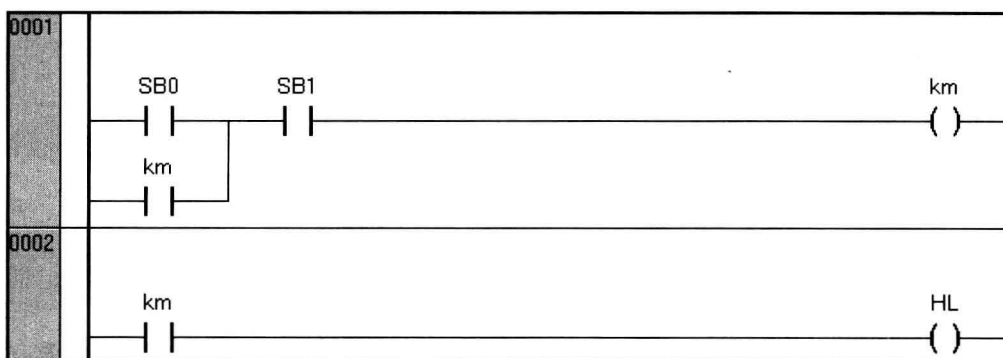


图 2.2 采用梯形图实现本例指示灯开关控制

变量声明部分与“2.1 指令表”中的变量定义相同。梯形图是 PLC 编程中最广泛使用的一种图形化编程语言, 来源于传统的继电器逻辑控制电路图。梯形图的编程图形元素与继电器线圈、继电器触点、控制按钮、电源轨线、连接线等电器元件具有一一对应的关系。PLC 控制器按照设定的控制周期由上至下逐行循环执行梯形图的控制语句。

以下是梯形图语言的编程元素。

(1) 触点元素

表 2.5 是梯形图触点元素的功能说明。

(2) 线圈元素

表 2.6 是梯形图线圈元素的功能说明。