

# 可编程序控制器编程语言标准 IEC 1131-3

上海交通大学 钟庆昌 谢剑英 陈应麟

**摘要:** 本文介绍了可编程序控制器编程语言国际标准 IEC 1131-3 的主要内容, 包括两种文本语言: 指令表 L 和结构化文本语言 ST, 两种图形语言: 梯形图语言 LD 和功能框图语言 FBD, 以及具有文本和图形两种表现形式的顺序功能表图 SFC 语言。

**关键词:** IEC 1131-3 可编程序控制器 编程语言 标准

## IEC 1131-3: Programming Languages for Programmable Controllers

Zhong Qingchang Xie Jianying Chen Yinglin

**Abstract:** This paper presents the international standard IEC 1131-3, which deals with the PC programming languages, including with two textual languages: Instruction List (L) and Structured Text (ST), two graphical languages: Ladder Diagram (LD) and Function Block Diagram (FBD), and the Sequential Function Chart (SFC) which can be represented both textually and graphically.

**Keywords:** IEC 1131-3 programmable controller programming language standard

## 1 引言

可编程序控制器(Programmable Controller)简称 PLC, 是一种以计算机的中央处理器为基础, 综合了电子技术、通讯技术和自动控制技术而开发的工业控制产品。它采用可编程序的存储器来存储执行逻辑运算、顺序控制、定时、计数和算术运算等操作的命令, 并通过数字式、模拟式的输入和输出, 控制各种类型的机械或生产过程<sup>[1]</sup>。

可编程序控制器于 60 年代末诞生于美国。当时, 美国最大的汽车制造商通用汽车公司 GM 为了适应汽车型号不断更新的需要, 公开招标需求一种新型的工业控制装置将计算机的灵活性、通用性与继电器控制系统的简单、方便、价格低廉结合起来。DEC 公司在 1968 年为此设计了第一台可编程序控制器。之后, 可编程序控制器以其优越的性能受到工业界的广泛重视, 各国相继研制了不同型号的可编程序控制器。到了 80 年代, 可编程序控制器发展成熟, 在工业控制领域占据了主导地位。我国在 70 年代末、80 年代初开始随设备引进了不少 PLC, 之后, 有许多科研单位开始研制和生产 PLC。目前, 在国内比较流行的可

编程序控制器有日本三菱及欧姆龙系列, 德国西门子的 S5、S7 系列, 瑞士 ABB 公司的 Master Piece 系列, 美国的 Rockwell Allen-Bradley 公司的 PLC-3、PLC-5 及 SLC 系列, AEG 施耐德公司的 MODICON 984 和 Quantum 系列, GE Fanuc 公司的 90TM-20、90TM-30 系列。

可编程序控制器在结构不断完善的同时, 编程软件也有了很大的发展, 由最初单一的梯形图(Ladder Diagram)语言发展到了有多种编程语言可供选用。然而, 各公司的编程语言都或多或少地针对自己的产品, 自成体系, 为用户的使用和系统集成带来了困难。

为了规范可编程序控制器的生产和应用, 国际电工委员会 IEC (International Electrotechnical Commission) 制定了可编程序控制器国际标准 IEC 1131, 成立了专门的推广组织 PLCopen<sup>[2]</sup>。在这一组标准中, IEC 1131-3 规定了可编程序控制器的编程语言。IEC 1131-3 一制定, 就得到了国际知名厂商的大力支持, 出现了以该标准为基础的 PLC 编程环境, 最典型的产品是 Wizdom Controls Inc 的 PARADYM-31。为了更好地促进可编程序控制器在我国的发展和应

用,使作为自动化三大支柱技术(PLC技术、机器人技术、CAD/CAM)之一的PLC技术为我国国民经济建设发挥更大的作用,本文详细介绍了IEC1131-3的基本内容。

## 2 IEC1131-3 标准简介

IEC1131-3标准由国际电工委员会于1993年3月颁布,分为四章、八个附录。第一章为概述,包括标准的范围,参照标准,术语定义,标准概览和一般要求,以及如何声明PLC系统及PLC程序符合该标准。第二章为通用元素,介绍PLC文本和图形编程语言的公共元素,包括字符的使用(含字符集、标志符与关键字的规定、空格的使用,以及如何使用注释)、数据的外部表示形式(数、字符串、时间)、数据类型、变量、程序组织单元(函数、函数块FB、程序)、顺序功能表图(Sequential Function Chart, SFC)元素、组态元素(用以实现向PLC安装程序)。第三章为文本语言,规定了两种文本编程语言:指令表(Instruction List, L)和结构化文本语言(Structured Text, ST),分别规定了其语法和语义。第四章为图形编程语言,规定了两种图形语言:梯形图(Ladder Diagram, LD)语言和功能框图(Function Block Diagram, FBD)语言。附录给出了文本语言完整的语法规则,据此可以开发出针对不同目标机且符合IEC1131-3标准的PC编译器以应用于不同的系统,从而实现最大程度的代码重用,减少开发时间。

## 3 PLC 编程语言<sup>[2]</sup>

### 3.1 文本编程语言

文本编程语言L和ST具有相同的类型声明语句以声明不同的数据类型、变量类型、函数、函数块、程序、状态步(step)、转移条件、动作等,关键字分别是:

```

TYPE ... END _TYPE  自定义数据类型
VAR ... END _VAR   声明内部变量
VAR _INPUT ... END _VAR  声明输入变量
VAR _OUTPUT ... END _VAR  声明输出变量
VAR _N_OUT ... END _VAR  声明I/O变量
VAR _EXTERNAL ... END _VAR  声明外部变量
FUNCT DN ... END _FUNCT DN  声明函数
FUNCT DN _BLOCK ... END _FUNCT DN _BLOCK
                                声明函数块
PROGRAM ... END _PROGRAM  声明程序
STEP ... END _STEP  声明状态步
TRANSIT DN ... END _TRANSIT DN  声明转移条件
    
```

ACT DN ... END \_ACT DN 声明动作

数据类型除了规定的基本数据类型(逻辑型BOOL、短整数SNT、整数NT、双字整数DNT、长整数LNT、无符号短整数USNT、无符号整数UNT、无符号双字整数UDNT、无符号长整数ULNT、实数REAL、长实数LREAL、时间段TME、日期DATE、时间TME\_OF\_DAY、日期时间DATE\_AND\_TME、字符串STRING、字节BYTE、字WORD、双字DWORD、长字LWORD)外,也允许用户自定义数据类型。

STEP、TRANSIT DN、ACT DN是SFC的3个元素,在3.3节详细介绍。

#### 3.1.1 指令表(Instruction List, L)

指令表L编程就是用一系列指令来完成特定的功能,每一条指令占一行,格式为:

标号:	指令码	操作数	(*注释*)
-----	-----	-----	--------

标准指令共有五大类:

```

数据传送:LD /LDN      取值/取反
              ST /STN    输出/求反输出
逻辑运算:S 置1
              R 清0
              AND /ANDN  与/与反(也可用 & / &N)
              OR /ORN   或/或反
              XOR /XORN  异或/异或反
算术运算:ADD 加法
              SUB 减法
              MUL 乘法
              DIV 除法
关系运算:GT  大于>
              GE  大于等于
              EQ  等于=
              NE  不等于< >
              LE  小于等于
              LT  小于<
    
```

转移指令:

```

JMP /JMPC /JMPN  无条件/当前值为1/为0转移
CAL /CALC /CALN  无条件/当前值为1/为0调用函数块
                  FB
RET /RETC /RETN  无条件/当前值为1/为0返回
    
```

运算指令的操作数可以是普通的数据类型,也可以是括号“( )”括起的操作数对,前括号紧跟指令码,后括号单独占一行。例如,指令组

```

MUL ( %IX1
ADD %IX2
)
    
```

表示的运算是:结果 = 结果 × (%IX1 + %IX2)。

除了标准指令码可以使用外,函数名也可以作为指令码来使用,函数的第一个参数是当前值,

其余参数在操作数域中给出,函数的返回值作为函数执行后的当前值。函数块的调用用 CAL 指令实现。

从上面的指令可以看出, L 编程比较简单,但不利于采用结构化的编程手段,不利于代码的重用,适合于较简单的系统。

### 3.1.2 结构化文本语言(Structured Text, ST)

结构化文本语言 ST 以语句为基本单位进行编程,每一语句以分号结尾。ST 的基本术语是表达式和语句,表达式是指由算符连接起来的操作数或函数,运算的优先级与常规的运算优先级一样,限于篇幅,此处从略。

ST 的语句包括:

赋值语句	变量 = 表达式;
函数块语句	函数块名(赋值参数列);
返回语句	RETURN;
条件语句	IF 条件表达式 1 THEN 语句 1; ELSE IF 条件表达式 2 THEN 语句 2; ... ELSE ... ENDIF;
选择语句	CASE 表达式 OF 值 1: ... ... 值 n: ... ELSE ... END -CASE;
循环语句	FOR/WHILE/REPEAT 三种形式
结束语句	EXIT; 用以终止循环语句。
空语句	;

ST 将计算机高级语言的许多功能吸收了过来,能够实现复杂的功能,程序可读性大大增强。

## 3.2 图形编程语言

### 3.2.1 梯形图(Ladder Diagram, LD) 语言

LD 语言是类似于继电器控制电路的编程语言,是应用最广泛的 PLC 编程语言。采用与 AND、或 OR 逻辑实现控制功能。IEC1131-3 允许采用功能块(在图形语言中,函数块称为功能块)来实现复杂的功能。LD 语言的主要元素是线、触点、线圈和功能块,前 3 者分别对应继电器控制系统中的连线、触点和线圈。

触点分为常开、常闭触点和上、下边沿检测触点。各触点的图形如下。

- 常开触点 ---- | |----或----! ! ----
- 常闭触点 ---- | / |----或----! /! ----

上升沿检测触点 ---- | P |----或----! P! ----  
检测到上升沿时置 1

下降沿检测触点 ---- | N |----或----! N! ----  
检测到下降沿时置 1

线圈分为如下几种:

普通线圈 --- ( ) --- 将当前值直接输出到线圈。

取反线圈 --- ( / ) --- 将当前值取反后输出到线圈。

置位线圈 --- ( S ) --- 若当前值为 1 就置线圈值为 1, 为 0 保持。

复位线圈 --- ( R ) --- 若当前值为 1 就置线圈值为 0, 为 0 保持。

保持线圈 --- ( M ) --- 具有掉电保护的普通线圈。

保持置位线圈 --- ( SM ) --- 具有掉电保护的置位线圈。

保持复位线圈 --- ( RM ) --- 具有掉电保护的复位线圈。

上升沿触发线圈 --- ( P ) --- 检测到上升沿时线圈置 1。

下降沿触发线圈 --- ( N ) --- 检测到下降沿时线圈置 1。

### 3.2.2 功能框图(Function Block Diagram, FBD) 语言

FBD 语言与电子电路设计中常用的框图相似,由具有一定功能的功能块 FB 组合而成,功能块之间由信号线连接。常用的功能块有与、或非逻辑块, R-S 触发器等,用户可以按照一定的格式

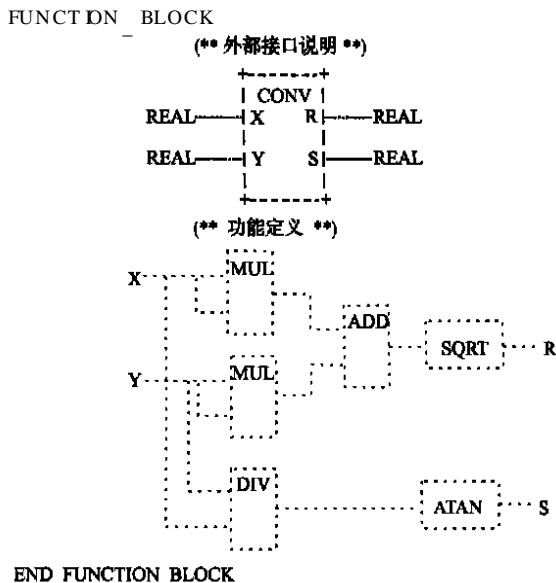


图 1 FBD 语言中功能块定义格式

自定义各种各样的功能块以实现不同的功能。已经声明的功能块可以作为定义新功能块的元素。因此,FB 语言编程就是定义功能块的过程。

下面以直角坐标(x, y)转换为极坐标(R, S)为例(x> 0),说明 FBD 语言中功能块的定义格式如图 1 所示。

在定义功能块 CONV 以后,就可以在别的地方按照外部接口说明的方式使用。

### 3.3 顺序功能表图(Sequential Function Chart, SFC)

SFC 编程有两种形式,既可以用文本方式,也可以用图形方式。SFC 编程是为了适应顺序控制而引入的。顺序控制就是将整个控制过程分为若干步(Step),每一步完成一定的动作(Action),当该步功能完成或者下一步的开始条件(Transition)满足时进入下一步。因此,SFC 编程的要素是:步(Step)、条件(Transition)、动作(Action)。限于篇幅,SFC 编程的详细方法从略。

## 4 结束语

本文详细介绍了可程序控制器编程语言国际标准 IEC1131-3 的主要内容,包括两种文本语言:指令表(Instruction List, L)和结构化文本语言(Structured Text, ST),两种图形语言:梯形图(Ladder Diagram, LD)语言和 功能框图(Function Block Diagram, FBD)语言,最后简要介绍了具有文本和图形两种表现形式的语言 SFC。

### 参考文献

- 1 田瑞庭. 可程序控制器应用技术. 北京:机械工业出版社, 1994
- 2 International Electrotechnical Commission. Programmable Controllers-Part 3: Programming Languages. 1993. 3

收稿日期:1998- 03- 31  
修改稿日期:1998- 06- 08

## 罗克韦尔与欧姆龙建立战略关系

这一举措将促进统一的技术和开放的标准,改造全球客户的工业自动化系统的性能

(2000 年 4 月 4 日,日本大阪) 自动化市场的龙头企业罗克韦尔自动化和日本欧姆龙公司今天宣布双方建立全球技术和产品合作关系。罗克韦尔自动化隶属于罗克韦尔国际公司,总部设在美国威斯康星州密尔沃基。

双方合作初期的重点将放在技术的开发、应用和升级,简化客户应用程序,改善工业自动化系统的性能。

这一合作是全球自动化用户的众望所归。欧姆龙首席执行官 Yoshio Tateisi 先生指出:“自动化用户期望供应商提供的产品更加广泛,功能更强大,性能更优良,同时他们期望自动化方案能够基于可互操作的开放标准。欧姆龙和罗克韦尔自动化的合作满足了双方的要求。两个公司通过在技术专长和自动化开放平台方面的联合,致力于为全球客户提供可互操作的自动化方案以及适用范围极广的高性能开放系统部件。”

首先,罗克韦尔自动化将向欧姆龙提供重要的技术协助,使欧姆龙的自动化平台能迅速采用罗克韦尔公司的开放的标准技术,如 DeviceNet, ControlNet 以及新近推出的 EtherNet。此外,欧姆龙将采取有力措施向其核心市场推广这些开放的标准技术。

其次,双方将共同合作通过 OPC 等开放技术以及罗克韦尔软件公司的自动化软件成分提高各自应用软件之间的连接性。典型的异构环境将使自动化用户提高生产力,降低集成成本,满足客户对互联网和车间之间透明地传输数据的要求。

罗克韦尔公司总裁兼首席执行官唐·戴维斯解释说:“虽然全球自动化市场在建立开放的自动化标准方面取得了相当的成就,然而地区差异仍然存在于这个销售总额达 360 亿美元的市场。因此全球用户仍然面临集成各个‘地区标准’的困扰。占市场主导地位的欧姆龙公司和罗克韦尔自动化在使用和推广统一的自动化技术方面达成一致,从而向减少地区障碍迈出了重要的一步。在亚洲,DeviceNet, ControlNet 以及 OPC 等开放的标准正逐渐得到认可,而来自欧姆龙这样重要公司的积极肯定将极大地增强用户乃至全球竞争者的信心。我们对由此将给两个公司带来的发展机会深受鼓舞。”

双方的合作将集中在技术协调,以便给自动化用户带来收益,并联合双方在研究方面的投资。罗克韦尔自动化和欧姆龙在进一步探索其他合作机遇的同时,将继续独立地在自动化市场开展业务,以各自独特的方式服务客户。

从长远来看,两个公司还将探索除联合技术开发以外的其他合作形式,包括产品互通以及各种形式的商业和操作手段如电子商务共享。欧姆龙首席执行官 Yoshio 补充说:“这种合作关系将带来行业内前所未有的竞争优势,有利于提高双方的业绩,促进公司发展。”

(罗克韦尔自动化市场部孟菲报导)