一种编制复杂 PLC 程序的新方法

The New Programming Method for Complicated PLC Programs

何　平[1]　兰生海[1]　麻万春[2]　梁座山[2]　赵玉珊[2]

（哈尔滨工业大学[1]，哈尔滨　150001 哈尔滨市科佳公司，哈尔滨　150001）

摘　要　介绍了一种新型的 PLC 编程方法并通过实例说明了具体的转换方法。利用该方法可以将 IEC61131－3 标准中的流程图在各种不同 PLC 硬件中实现编程，从而实现了流程图编程的硬件无关性。该方法方便了各种 PLC 在工业控制领域的应用。

关键词　PLC　工业控制　流程图编程

Abstract　The new PLC programming method is presented. Through practical example the concrete conversion method is illustrated. By using this method, the program applied in various hardware of PLC can be implemented from the flowchart conforming IEC61131－1 standard. Thus the flowchart programming independent on hardware is implemented and it will be easier for PLC application in industrial areas.

Keywords　PLC　Industrial control　Flowchart programming

0　序言

虽然 IEC1131 规定有 SFC（即顺序控制）表达方法，但对于许多中小型 PLC，特别是我国目前大量使用的 PLC 早期产品，一般都没有 SFC 功能。目前厂商免费提供的编程软件，一般都是通过精简以后的，功能一般都压缩到最小，仅仅可以用梯形图或指令表来实现编程。如果遇到需要设计复杂控制逻辑的应用系统，其编程难度会非常大。为了解决这种问题，有人曾设计出专用的 PLC 编程系统，通过软件将逻辑式转化为梯形图。但是这种软件仅限于东芝和欧姆龙系列的 PLC，并且不能提供源代码，对其它型号的 PLC 根本无法应用。为此，本文给出一种比较系统的方法来实现由流程图向梯形图的转化，但是目前这种方法还必须通过手动来实现，没有相应的软件支持。

1　针对不同 PLC 硬件的转化方法

在转化之前，必须确定该 PLC 支持那些指令集。如果硬件支持步进指令（SFC 指令）的话，转化起来就相对容易。如果不支持步进指令，但支持跳转指令，那么转化起来就相对复杂一些。如果这两种指令都不支持，那么实现起来就更加复杂。

1.1　PLC 硬件支持步进指令

在转化之前，首先必须画出系统的流程图。有了系统的流程图就可以进行转化了，其主要步骤如下：

① 将普通的顺序流程图进行规划，将可以同时执行的动作尽可能合并在一个模块中，以减少流程图中的模块，降低程序的复杂性。如控制一个绘图仪，将笔头移动到（x，y）坐标上，这时先沿 x 轴移动还是先沿 y 轴移动对最终的结果都没有影响，此时我们就可以将这两个动作合并在一个模块中。

② 为每一个模块分配序号。每个模块的标志就是这些序号了。每一种 PLC 都有其可用的序号范围，并不是无限大。例如，松下的 FP sigma 的序号范围为 0～999，即最多允许 1000 个程序模块。

③ 实现转化。为每个模块的开始加入步进梯形图的开始标志，其序号就是在第二步分配的；在本模块的结尾加入步进梯形图的结束标志。将本模块要实现的代码放入其中，在需要实现状态转移的位置加入步进梯形图的转移指令，转移参数就是下一个模块的序号。

图 1 为一个典型的流程图，共有 7 步，也就是 7 个模块，包括选择模块（如 S_1 和 S_2）和并行模块（如 S_3 和 S_4）等。首先，对这些模块进行标号，即图中的 S_0～S_6，并标出转化条件，如图中的 R_0～R_5。接下来就可以进行转化了（为了简单起见，以下例子中的梯形图指令都是利用松下 PLC 的指令集实现的）。

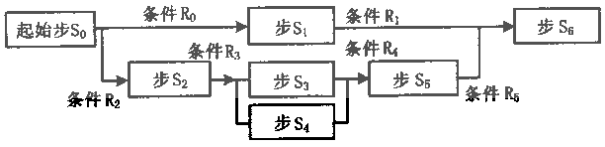

图 1　流程图

首先转化模块 S_0。转化后的梯形图如图 2 所示，SSTP 指令为步进梯形图的开始标志，后面的数字为步进梯形图的步进序号。STPE 为步进梯形图的结束标

志。指令 NSTL 可以实现几个步进梯形图之间的切换。在本模块中，如果条件 R_0 满足的话就转到模块 1 如果条件 R_2 满足的话则转入模块 2。但是必须人为地保证 R_1 与 R_2 不能同时满足。

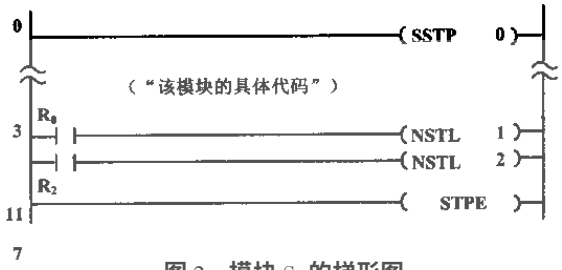

图 2　模块 S_0 的梯形图

标准的顺序模块 S_1 转化后的梯形图如图 3 所示。

图 3　模块 S_1 的梯形图

用来触发同步模块的 S_2 模块转化成梯形图后如图 4 所示。主要原理是通过一个条件继电器 R_3 来触发两个模块 S_3 和 S_4。

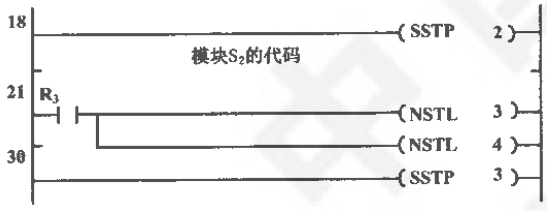

图 4　模块 S_2 的梯形图

同步模块 S_3、S_4 的转化结果如图 5 所示。为了使编写的程序模块具有独立性，我们又引入了两个中间寄存器 R_6、R_7。R_7 记录模块 S_3 的执行结果，如果闭合则认为模块 S_3 已经执行完毕。R_4 则记录模块 S_4 的执行结果。这样如果 S_3、S_4 均已闭合，说明模块 S_3 和模块 S_4 都已经执行完毕，可以进入下一个模块。此时，通过 R_7 和 R_4 的与逻辑来触发下一个模块，即模块 S_5。

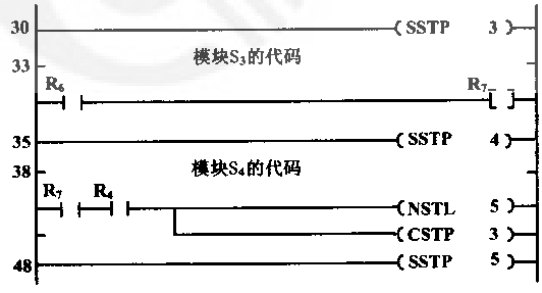

图 5　模块 S_3、S_4 的梯形图

但是在进入下一个模块之前一定要将同步模块的执行标志清除。这样，当下一步扫描周期到达的时候就不行执行模块 S_3、S_4，而只执行 S_5。在 FP sigma 指令集中，可通过 CSTP 清除模块的执行标志，参数即为模块的标号。在本例中，利用 CSTP 3 指令清除 S_3 的执行标志，利用 NSTL 清除本模块（即 S_4）的执行标志。

其他模块的转化过程和以上几个模块的转化过程类似，就不一一列举了。

1.2　PLC 硬件支持跳转指令

在一些早期的 PLC 中，一般都不支持 SFC 指令。在此，我们充分利用跳转指令和状态转移图的原理来模拟 SFC 指令，以达到相同的执行效果。

首先，将程序的流程图转化为状态转移图。例如可以将流程图 1 转化成状态转移图 6。为了更真实地反映 PLC 的执行过程，在流程图的基础上加入了返回第一步的条件 R_8 和自循环，从而更加符合 PLC 程序的执行规律（扫描特性）。在图 4 ~ 8 中，没有写转换条件的转化路径为状态的自循环，即没有触发下一个状态的情况下程序始终在本状态运行。从图中可以发现，利用状态转移图非常清晰地描述出了 PLC 实际执行状况，有助于初学者编写 PLC 程序。由于状态转移图中处于激活的状态只有一个，所以对于前面提到的并行模块，必须将它们合并为一个状态，如 S_3、S_4 状态。

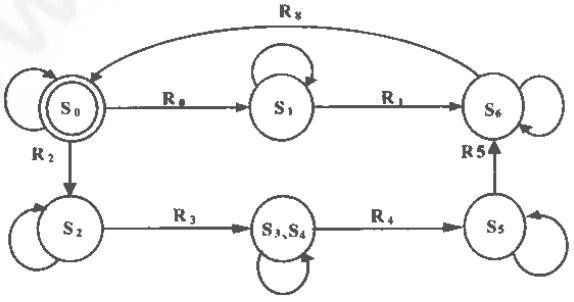

图 6　状态转移图

其次，将每个状态所有的梯形图代码分别书写在一起，为每一个状态分配的唯一的标识符写在该段代码的入口处，并在整段代码的最后书写跳转指令，统一跳到程序末尾。当要跳入下一个状态时，清除本状态的执行条件，设置下一个状态的执行条件。

最后，书写状态转移图的入口代码和结束代码。入口代码负责整个状态转移图的驱动，如当前该进入哪一个状态 结束代码可以仅仅是一个标号，起到标识状态退出点的作用。

图 7 为状态转移图 6 转化成梯形图后的部分梯形图程序。由于编程软件 FPWin GR 的跳转标志只能为数字，因此所有的模块标志都是通过数值来表示的。

其中第 0 步到 21 步为状态转移图的入口代码。第 21 步中的 R9010 为常闭继电器，当程序执行到第 21 步的时候一定跳转，从而起到保护程序的作用，防止发生误执行。第 25 步到 41 步之间为模块的具体代码，如第 25 到 33 为模块 0 的梯形图代码。第 41 步为结束代码，所有模块执行完之后都跳转到第 41 步，方便后面程序的编写。模块之间的切换通过 SET、RST 指令实现。当要退出本模块进入下一个模块的时候，将本模块的触发继电器（如模块 0 的触发继电器为 R_8）复位，将下一个模块的触发继电器（如模块 1 的触发继电器为 R_0）置位。

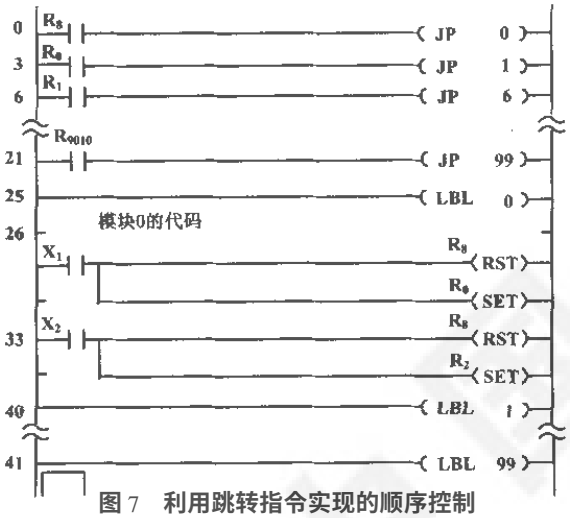

图 7　利用跳转指令实现的顺序控制

1.3　PLC 硬件对前两种指令都不支持

在这种情况下实现顺序控制比较繁琐，结构不是很清晰。而且如果 PLC 硬件对这两种指令都不支持的话，说明这个 PLC 的型号已经很落后了，功能也一般很弱，所以不适合实现复杂控制。

把前一种方法经过简单的改造就可以实现在不使用跳转指令的情况下的顺序控制。这种方法是充分利用继电器的通断特性来实现的。

首先，将程序流程图转化成状态转移图（具体参见第二种方法）。

然后，分配触发继电器。通过这些继电器来标识当前哪一个模块正在执行，并且通过断开和闭合这些继电器来完成模块之间任务的切换。

最后，利用分配好的触发继电器来完成程序的编码。

图 8 就是前面例子的部分实现代码。由图 8 可以看出，模块 0 的代码必须全部写在触发继电器 R_8 的后面，这样虽然对编程造成了修改起来不太方便的弊端，但是整体结构还是比较清晰的。如果触发继电器 R_8 闭合，那么模块 0 的代码被执行（第 0 步到 32 步）。如果触发继电器 R_0 闭合，那么模块 1 的代码被执行（第 32 步以后的代码）。模块之间的过渡方式和上一种方式相似，通过断开本模块的继电器、闭合下一个模块的继电器来实现。必须注意，当 R_8 断开时，模块 0 的代码并不是不执行了，而是由于任何一个继电器和一个断开的继电器相与，结果都为断开而已。

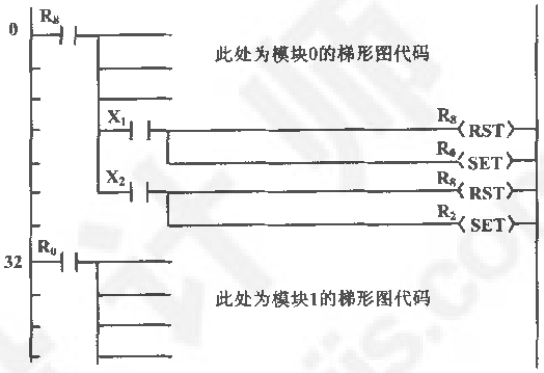

图 8　利用触发继电器实现顺序控制

2　结论

利用本文提到的转化方法可以很容易地将工业中常用的顺序控制通过任何一种 PLC 来实现。因此，我们可以在不增加硬件成本和软件成本的基础上实现各种复杂的控制功能。

在这三种方法中，第一种实现起来最为容易，但是支持步进指令的 PLC 价格比较高。最后一种实现起来比较复杂，而且不是真正意义上的顺序控制，所以不推荐使用。

参考文献

1　Hu W.，Schroeder M.，Starr A. G. A knowledge-based real-time diagnostic system for PLC controlled manufacturing systems. IEEE SMC '99 Conference Proceedings，1999(4)

2　Clarke S. J. Safety considerations in using IEC 1131 - 3 programming languages. Open Control in the Process and Manufacturing Indusgtries，1998(5)

3　IEC - 1131 - 3. 编程语言. 1993(3)

4　崔亚军. PLC 梯级图自动编程语言与系统. 机床电器，1995(1)

5　Krairojananan T.，Suthapradit S. A PLC program generator incorporating sequential circuit synthesis techniques. Circuits and Systems，IEEE APCCAS 1998. 1998(12)

6　Matsushita Electric Works，Led. FP series Programmable Controllers Programming Manual.

7　向树民，段晨东. PLC 全变量编程方法. 重型机械，1997(5)

收稿日期：2003 - 12 - 20。

第一作者何平，男，1962 年生，毕业于哈尔滨工业大学，获博士学位，教授：主要研究领域：飞行器导航、制导与控制以及现代测量与现代控制技术。