

文章编号: 1002-0411(2000)01-047-09

机床控制流程的一种有限状态机表达方法

雷为民¹ 于东² 李本忍² 滕弘飞¹

(1. 大连理工大学机械工程系 116024; 2. 中国科学院沈阳计算技术研究所, 高档数控国家工程研究中心 110003)

摘要: 面向过程的 IEC-1131-3 规范已难以满足机床控制流程表达新的应用需求, 为了更好地支持控制器开放式体系结构设计和面向对象系统实现技术, 我们扩展了有限状态机的基本概念, 提出了一种机床控制流程表达的分层式有限状态机(FSM)方法. 本文首先对分层式 FSM 的组织方法、特性、形式定义等进行了详细讨论; 为了进一步阐明这种方法的表达特性, 我们介绍了一种分层式 FSM 表达的机床控制器总体结构, 并讨论了这种结构下的开放性设计表达和系统实现等相关问题。

关键词: 机床控制器, 控制流程表达, 分层式有限状态机, IEC-1131-3 规范

中图分类号: TP391

文献标识码: B

1 引言

在机床控制系统的规划、设计与实现中, 控制流程的设计表达是一个重要的环节, 有效的设计表达不仅可以简化系统的复杂度、促进系统的快速实现、充分体现系统体系结构思想, 而且可以通过这种形式主义的表达来验证控制流程的合理性, 排除控制系统设计中的各种错误。

目前在一般控制系统设计中, 比较常见的控制流程表达工具是 IEC-1131-3 规范^[2], 在机床控制器设计(特别是控制器中的 PLC 设计)中, IEC-1131-3 具有广泛的应用基础。

机床控制系统正在经历着新的技术变革, 支持开放式体系结构的系统设计和面向对象的系统实现已逐渐成为机床控制器设计和实现技术的主流. IEC-1131-3 规范以结构化、模块化过程分析方法为基础, 是一种面向过程的表达工具, 已难以适应这种新的应用需求。

有限状态机(Finite State Machine, 简称 FSM) 又称有限自动机(Finite Automaton, FA), 是一种具有离散输入输出系统的数学模型, 它以一种“事件驱动”的方式工作, 可以通过事件驱动下系统状态间的转移, 来表达一个控制系统的控制流程。

有限状态机和面向对象程序设计具有相似的工作方式, 其基本特性又非常适合表达动态控制行为, 我们认为在有限状态机的概念基础上加以扩展, 可有效地表达新型机床控制器的控制流程设计。

2 有限状态机基本概念

有限状态机是由状态的有穷集和状态之间的转移关系组成. 其形式定义如下。

定义 FSM 是一个五元组,

$$M = (Q, \Sigma, \delta, q_0, F), \text{ 其中}$$

(1) Q 为有穷状态集;

收稿日期: 1999-04-19
基金项目: 中国科学院机器人学开放实验室基金课题(A970111)

- (2) Σ 有穷的事件集;
 (3) δ 为从 $Q \times \Sigma$ 到 Q 上的映射或称转移函数;
 (4) q_0 在 Q 中, 它是初始状态;
 (5) $F \subseteq Q$ 为终止状态集.

FSM 具有状态转移表、状态转移图等多种描述方式.

状态转移图是一种形象、直观的 FSM 描述方式, 它是一个有向图, 有向图中的顶点对应于 FSM 中的状态; 顶点间的路径对应于 FSM 中与顶点对应的状态间由于某特定事件 $e (e \in \Sigma)$ 引起的状态转移; FSM 中初始状态和所有的终止状态 $q_{f1} \dots q_{fn} (q_{f^*} \in F)$ 通常对应于具有特定标记的顶点.

例如, 下列有限状态机描述的是机床控制中某一局部的控制流程. 其 FSM 的完全表示为:

$$FSM = (Q, \Sigma, \delta, q_0, F)$$

其中, $Q = \{ \text{系统就绪, 轴运动, 刀具破损处理, 系统停机} \}$; $\Sigma = \{ \text{加工请求, 轴进给, 进给完成, 刀具破损, 人工换刀} \}$; δ 包括: $\delta(\text{系统就绪, 加工请求}) = \text{系统就绪}$; $\delta(\text{系统就绪, 轴进给}) = \text{轴运动}$; $\delta(\text{轴运动, 进给完成}) = \text{系统就绪}$; $\delta(\text{轴运动, 刀具破损}) = \text{刀具破损处理}$; $\delta(\text{刀具破损处理, 刀具就绪}) = \text{系统就绪}$; $\delta(\text{刀具破损处理, 人工换刀}) = \text{系统停机}$; $\delta(\text{系统就绪, 加工结束}) = \text{系统停机}$; $q_0 = \text{系统就绪}$ $F = \{ \text{系统停机} \}$

其对应的状态转移图见图 1.

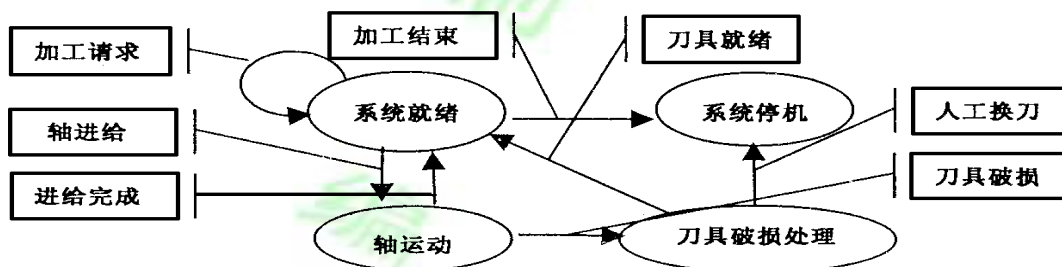


图 1 机床控制中某一局部的 FSM 状态转移图示例

3 用于机床控制流程表达的分层式 FSM

3.1 机床控制的基本特征

机床控制的特征直接影响控制流程的表达. 根据 FSM 的表达特性, 进行控制流程的系统规划时, 要充分地考虑这些特征.

1) 复杂的串级控制

机床控制被认为是一种复杂的串级控制, 其主要控制流程通常被划分为三个层次: 工艺控制、运动控制/逻辑控制、伺服控制/设备控制. 三个层次的控制流之间同时具有顺序和并发的关系.

2) 多闭环系统

机床控制通常具有多级反馈处理机制, 是一种典型的多闭环系统.

3) 控制策略的开放性需求

随着开放式体系结构的发展, 控制策略的可替换性、可修改性, 也逐渐成为机床控制系统的基本特征. 这要求 FSM 表达的控制流程具有较强的“可配置性”或者是一种“自生成系统”.

4) 智能控制支持需求

智能控制技术正逐渐渗透到机床控制中, 所以对机床控制流程的表达要考虑对模糊控制、神经网络控制等控制策略的支持.

3.2 分层式 FSM 及其组织方法

根据机床控制的基本特征, 我们提出一种分层式 FSM 表达方法. 分层式 FSM 的组织类似于软件工程中结构化分析方法中的数据流图(DFD). 其组织原则是: 1) 状态和层次的划分机床控制是一个复杂控制系统, 其控制行为可划分为多个相互协作的控制功能群, 每个控制功能群又被划分为多个相互协作的控制功能子群, 不能被细分的控制功能子群称为控制基元. 见图 2 示意.

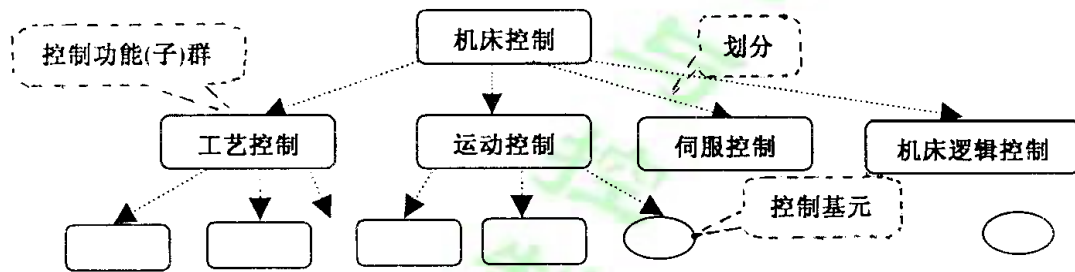


图 2 机床复杂控制系统控制功能划分示意图

在分层式 FSM 中, 每个控制基元对应着一个 FSM 状态; 多个控制基元组成的控制功能子群分别对应着一个 FSM 及一个上层的 FSM 状态; 控制功能(子)群间的层次和逻辑关系构成了相应 FSM 间的层次和逻辑关系. 这样, 机床控制系统的控制流程可由这样一组 FSM 来表达: {一个顶层 FSM, 若干个一层 FSM, 若干个二层 FSM, ...}. 图 3 示意了分层式 FSM 组织中的状态和层次关系.

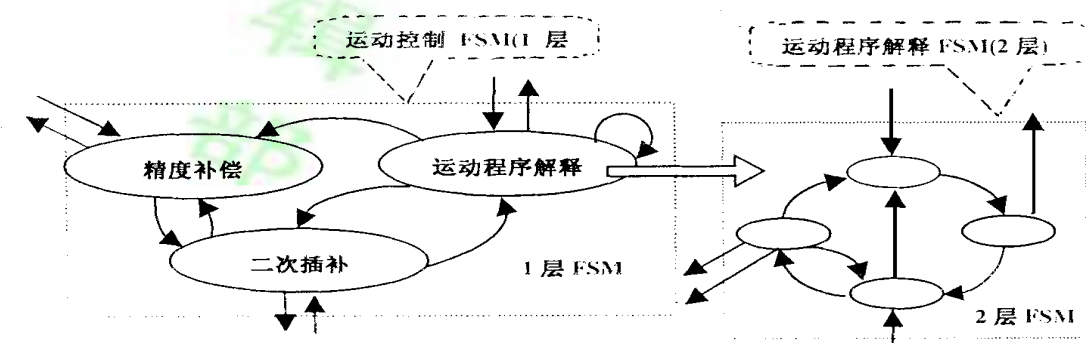


图 3 分层式 FSM 组织中的状态和层次关系示意图

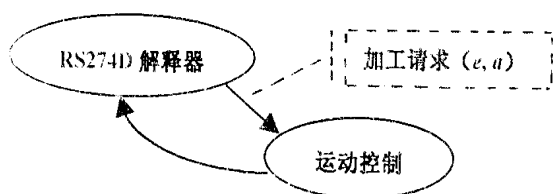
2) 事件

FSM 通过对事件的响应、实现其状态转移, 从而表达控制流程的演变. 由于机床控制流程

的三个层次间同时具有顺序和并发的关系, 分层式 FSM 采用一种支持并发的顺序事件模型. 事件按照其来源, 被分为两类: 系统事件和用户事件. 与反馈相关的事件通常被设定为系统事件(或称外部事件), 其他事件均被设定为用户事件(或称内部事件), 在系统实现中, 系统事件和用户事件被分别组织成两个事件队列, 在并发 FSM 中, 对系统事件的响应通常具有更高的优先级.

为了更形象地表达分层式 FSM 的层次关系, 根据事件对应的目标状态的形式, 事件被分为两类: 分枝事件和叶子事件, 如果事件对应的目标状态可被分解为一个低层 FSM, 则称该事件为分枝事件, 如果目标状态是不可再分解的, 则称这个事件为叶子事件.

在机床控制流程的 FSM 表达中, 事件是一个二元组 (e, a) , “ e ”代表事件的触发器, “ a ”是



指执行一种控制行为. 见图 4 示例.

* 这里 e = 一次插补周期完成;

a = 某个函数(): 其主要功能是: 将对各运动轴插补结果或生成的运动控制程序传递给运动控制部件.

* 这个事件是一个分枝事件.

* 这个事件是一个用户事件.

图 4 机床控制的分层式 FSM 表达中的事件

3.3 事件、状态和层次间的协调

任何分枝事件总是与一个代表同一事件的叶子事件对应, 分枝事件只是这个叶子事件在上层 FSM 中的表达形式.

同一 FSM 中状态间的转移是由事件驱动的, 同一层次的不同 FSM 之间通常没有通信问题, 如果这些不同 FSM 中的状态所表达的控制流具有并发的特性, 它们之间的同步是由上层同一 FSM 中的状态来协调的.

在机床控制流程的分层式 FSM 表达中, 一般总是上层 FSM 协调其相关的下层 FSM 或状态间的控制行为; 另外, 机床逻辑控制 FSM 也起着重要的协调作用, 机床逻辑控制 FSM 通过其底层 FSM 向其他 FSM 发送事件并收集状态, 来有效地协调系统的控制流程.

上层 FSM 对下层 FSM 间的协调, 引发了上下层 FSM 间的通信问题. 为了支持这种通信, 在分层式 FSM 的实现中, 每个 FSM 都有一组通信端口, 同层的不同 FSM 间共享通信端口, 但同时只能有一个 FSM 响应某一特定事件. 为了引用一个低层 FSM, 上层 FSM 发送一个事件到下层 FSM 的通信端口; 当下层 FSM 到达终止状态时, 发送一个终止通知事件到上层 FSM 的通信端口.

图 5 中, 2 层 FSM 是 1 层 FSM 中“轴控制”状态对应的低层 FSM, 两层 FSM 间的通信是通过相应的通信端口和一个公共的通信对象完成的.

3.4 FSM 的管理属性

分层式 FSM 由位于不同层次相互关联的多个 FSM 组成, 其中, 每个 FSM 都有一组独立的管理属性, 包括:

(1) 带有层次和关联信息的唯一标识

任何 FSM 状态都有一个全局性的 ID, ID 是一个分节的字串, 如“2.3.1”等, 其含义为:

第一节为‘2’, 表示在顶层 FSM 中与状态 2 相对应; 第二节为‘3’, 表示在相应的 1 层 FSM 中与状态 3 相对应; 末节为‘1’, 表示在本 FSM 中, 状态编号为‘1’.

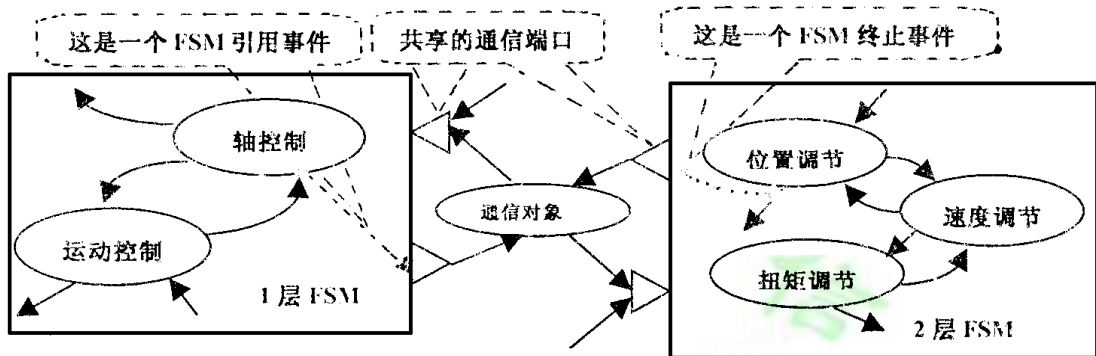


图 5 不同层次 FSM 间的通信机制示意图

FSM 的标识和与其对应的上层 FSM 中的状态的 ID 是完全一致的。

(2) 可替换性标志

一个布尔类型值. 它标识在控制器生成时, 这个 FSM 是否可被具有同样功能接口的其他 FSM 所替换.

(3) 可删除性标志

一个布尔类型值. 它标识在控制器生成时, 这个 FSM 是否可被删除. 如果这个 FSM 是系统控制流程中的“关键路径”, 那么它是不可删除的.

(4) 外部引用事件集合

引用这个 FSM 的所有事件的集合. 它和终止事件集合一起构成了 FSM 功能接口.

(5) FSM 终止事件集合

如果一个事件触发的转移函数的目标状态不在这个 FSM 之内, 我们称这个事件为 FSM 终止事件. 终止事件集合是 FSM 功能接口的一部分.

管理属性的定义, 使得这种改进的 FSM 不仅可以更灵活地表达机床控制流程, 而且是控制器可配置性设计的基础.

3.5 分层式 FSM 的形式定义

分层式 FSM 是根据机床控制系统的基本特征和控制流程表达的需求, 在传统有限状态机概念的基础上扩展而成的一种有限状态机系统. 按照我们对其基本特征的讨论, 其形式定义可用巴克斯范式表示为:

分层式 FSM ::= < 多层 FSM > + < 全局通信对象 >

< 多层 FSM > ::= < 单层 FSM > + < 单层 FSM 通信接口 > + [{ < 单层 FSM > + < 单层 FSM 通信接口 > }]

< 单层 FSM > ::= < 单元 FSM > + [{ < 单元 FSM > }]

< 单元 FSM > ::= < FSM > + < 管理属性集 >

< FSM > ::= < 状态集 > + < 事件集 > + < 转移函数集 > + < 初始状态 > + < 结束状态集 >

< 状态集 > ::= < 状态 > + [{ < 状态 > }]

< 状态 > ::= < 单元 FSM > | < 元状态 >

4 分层式 FSM 表达机床控制流程的几个相关问题

4.1 控制器总体结构

控制流程的设计表达要直接面向系统的实现, 根据机床控制流程的特点, 我们设计的控制器总体结构可表达为: 见图 6.

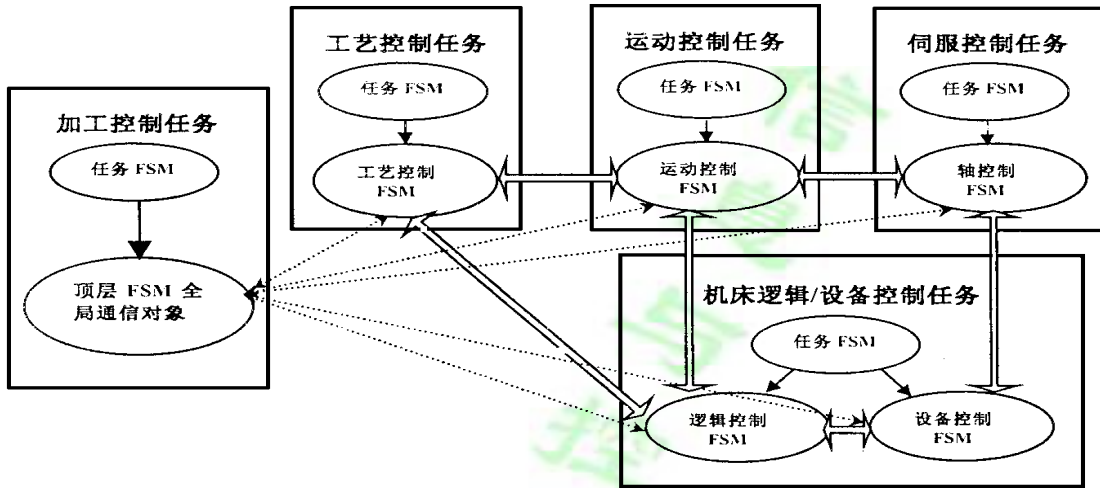


图 6 分层式 FSM 表达的机床控制器总体结构

在这个设计表达中, 控制器由多个可以并发的任务组成, 每个任务包含一组耦合较强的控制流程, 完成一组特定的控制功能. 在图 6 中, 加工控制任务对应着顶层 FSM 表达; 其他任务分别对应着某个 FSM 状态, 并一起构成了一层 FSM 表达; 一层 FSM 中的状态转移, 是通过任务间的通信(全局通信对象, 可用共享内存等机制实现)实现的.

4.2 控制器的开放性设计表达

控制器的开放性设计表达主要体现在两个方面.

(1) 任务 FSM 的引入, 使得控制系统具有增强的可配置性.

我们注意到在图 6 中, 每个任务中都包含两种 FSM: 一个“任务 FSM”; 一个或两个“功能 FSM”. 这里, “任务 FSM”的主要功能是管理本任务中“功能 FSM”的生成、配置和执行. 其状态转移图见图 7.

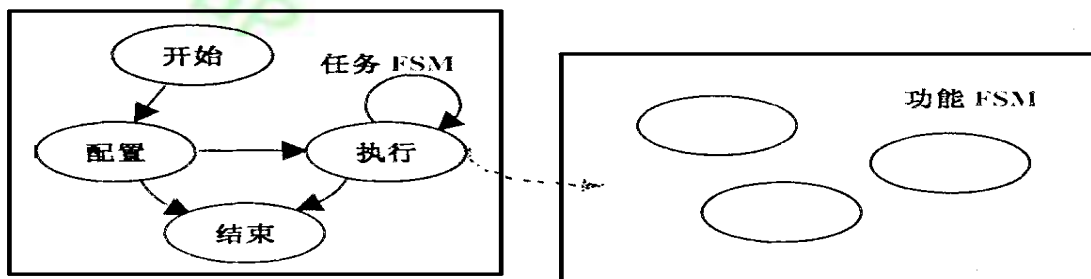


图 7 任务 FSM 的状态转移图

任务 FSM 在配置状态下, 将根据用户需求和机床特性组合相应的功能 FSM 部件, 并对其各种参数配置。

(2) FSM 管理属性是可配置性表达的基础

机床控制流程的 FSM 表达中, 所有的 FSM 被称为系统元件, 如果两个元件具有同样的功能接口, 并且其管理属性中的可替换性标志均为真, 则这两个元件被称为是“相似”的。在控制生成和配置时, 相似的系统元件是可以被相互替换的, 这将有助于控制策略的开放性实现。如, 对于“轴控制 FSM”之下的“速度调节 FSM”, 可以有“基于 PID 的速度调节 FSM”、“基于模糊控制的速度调节 FSM”、“基于用户定制的速度调节 FSM”等多个相似的系统元件。对相似系统元件的选定并不影响控制系统的整体结构, 具有相似性的系统元件所表达的控制流程粒度越小, 控制系统的开放程度将越高。另外, FSM 的可删除性, 直接支持系统为适应机床规模特性而进行的参数调节。

FSM 的功能接口、相似性判断、可替换性判断、可删除性判断等均依赖于 FSM 管理属性中的信息, 所以说, FSM 管理属性是系统可配置性表达的基础。

4.3 设计表达与系统实现

有限状态机的基本工作方式是“事件驱动”, 这和面向对象程序设计有本质的一致性。分层式 FSM 设计是直接面向系统实现的, 将系统设计转化为系统的面向对象实现时, 可采用下列规则:

- (1) 相似的 FSM 系统元件的共性被抽象为类, 其特性体现在类的派生;
- (2) FSM 的管理属性是类的属性;
- (3) 分层式 FSM 中的任何一个 FSM (不论层次) 均对应一个具体的对象;
- (4) 所有的 FSM 状态均对应一个具体的对象, 状态对象是其相应 FSM 对象的公有成员;
- (5) 一个 FSM 和其对应的上层 FSM 状态对应着同一个对象;
- (6) 同层 FSM 通信接口是同层 FSM 类的静态成员函数;
- (7) 系统通信对象类是所有 FSM 类的友元;
- (8) 事件的触发对应着源状态对象类成员函数, 事件的响应对应着目标状态对象类成员函数;
- (9) 实现系统事件队列和用户事件队列, 由一个队列管理对象进行管理;
- (10) FSM 状态转移函数是通过事件触发、事件管理、事件响应所对应的三个不同的对象成员函数实现的。

5 评价和结论

机床控制流程的分层式 FSM 表达设计, 是我们 BS-2000 软件数控^[1]研究项目的一部分, 这个研究项目的主要内容包括两个方面: (1) 完全用 PC 应用软件的形式实现机床控制中的伺服控制; (2) 机床控制策略的用户开放性实现方法。在项目研究中我们发现, 机床控制流程的表达是一个非常重要的问题, 有效的设计表达, 不仅有助于我们迅速掌握机床控制流程的主要特征, 抓住系统实现的主要问题, 有效地支持系统的开放式体系结构, 而且可以简化系统的复杂程度, 有利于控制原型系统的快速实现。

IEC-1131-3 规范是由国际电工协会(International Electrotechnical Commission)于 1993 年推出的一个控制流程表达工具, 这个规范实际上是一种主要面向可编程控制器(PLC)的程

序设计语言,其主要内容包括用于表达程序结构的有序功能图(Sequential Function Charts, SFC)及四种操作语言:指令列表(Instruction List, IL)、梯形图(Ladder Diagram, LD)、功能块图(Function Block Diagram, FBD)和结构化文本(Structured Text, ST). IEC-1131-3 规范具有较强的流程描述能力,并且支持系统的可配置性设计,在控制系统设计中获得了广泛的应用.

但随着相关技术的进步,IEC-1131-3 规范的局限性已阻碍了它的深入应用.其局限性主要表现在:

(1) 与面向对象程序设计具有较大的语义差距

IEC-1131-3 规范强调面向过程的结构化设计,是一种过程程序设计语言,与面向对象语言具有较大的语义差距.而目前,面向对象已成为控制系统实现的主流技术,这种较大的语义差距使得控制系统的设计和实现难以统一,增加了系统实现的难度.

(2) 可配置性设计支持不够充分

IEC-1131-3 规范通过支持结构化控制模块、变量参数方式等支持系统的可配置性设计,随着开放性需求的增长,可配置性设计要求支持“控制策略的用户定制”等更深入的开放性,IEC-1131-3 规范已难以满足这样的需求.

为了克服上述局限性,我们在有限状态机概念的基础上加以扩展,并借用了软件工程中数据流图、软件数控参考模型^[1]中的管理层面等分析方法和基本思想,提出了用以描述机床控制流程的分层式 FSM 方法.

有限状态机“事件驱动”的工作方式与面向对象程序设计的“消息驱动”是完全一致的,它们有非常相似的语义.有限状态机基于“事件驱动”的“状态转移”也非常适合动态控制流程的表达,分层式 FSM 进一步增强了这种表达能力,不仅通过层次划分和相关通信机制的引入,支持复杂控制流程的表达,而且通过在 FSM 上附加管理属性,更充分地支持机床控制系统的可配置性设计.

我们采用分层式 FSM 方法设计了 BS-2000 机床控制器的部分控制流程(运动控制的一部分),并用 C++ 进行了系统实现,可以认为这是一种非常有效的机床控制流程表达方法.在实验中我们还发现,分层式 FSM 虽然采用一种支持并发的顺序事件模型,但对复杂系统中并发控制流称的表达并不是特别有效,这是这种方法有待改进的地方.

参 考 文 献

- 1 雷为民,乔建中,李本忍,滕弘飞.关于软件数控的一些基本构想.小型微型计算机系统,1999,20(2):81~87
- 2 PLC Open Association. The IEC 1131-3 standard, <http://www.plcopen.org/stdscr.htm>
- 3 Chito Shiu, Michael J Washburn, Shige Wang, Chinya Ravishanker. Specifying Reconfigurable Control Flow for Open Architecture Controllers. Proc. 1998 Japan-USA. Symposium on Flexible Automation, Ohtsu, Japan, July 1998: 659~666
- 4 S K Birla. Software Modeling for Reconfigurable Machine Tool Controllers. Ph. D. Dissertation, The University of Michigan, Ann Arbor, MI, May 1997
- 5 邹海明,周新.形式语言、自动机和语法分析.武汉:华中工学院出版社,1985
- 6 Harel D. State Charts: A Visual Formalism for Complex Systems. Science of Computer Programming, 1987, 8(3): 231~274

(下转第 75 页)