

引文格式: 王兵, 江海凡, 丁国富. 基于 DEVS 的柔性制造系统逻辑建模与 PLC 程序设计 [J]. 航空制造技术, 2021, 64(8): 41-49.
WANG Bing, JIANG Haifan, DING Guofu. Logic modeling and PLC program design based on DEVS for flexible manufacturing system[J]. Aeronautical Manufacturing Technology, 2021, 64(8): 41-49.

基于 DEVS 的柔性制造系统 逻辑建模与 PLC 程序设计*

王 兵, 江海凡, 丁国富

(西南交通大学机械工程学院, 成都 610031)

[摘要] 针对柔性制造系统控制方案设计过程中逻辑复杂、调试周期长、调试成本高等问题, 提出一种基于可试验数字孪生体的虚拟调试方法。该方法以分层的方式建立物理系统的孪生模型, 基于离散事件系统规范的扩展形式构建设备层、单元层、系统层 3 层的逻辑模型, 通过仿真对逻辑模型进行重复验证与修改。建立了逻辑模型到 PLC 程序的映射规则, 实现了 PLC 程序的快速设计。最后, 以某柔性制造系统为例, 详细演示了逻辑建模过程及 PLC 程序的设计, 验证了所提方法的可行性和有效性。

关键词: 柔性制造系统; 可试验数字孪生体; 虚拟调试; 逻辑建模; 离散事件系统规范; 可编程逻辑控制器

DOI: 10.16080/j.issn1671-833x.2021.08.041



王 兵

硕士研究生, 主要研究方向为数字化车间仿真、虚拟调试。

随着产品更新换代周期日益缩短以及产品复杂程度的不断增加, 柔性制造系统(Flexible manufacturing system, FMS)^[1]需要满足更多、更高的要求, 例如更短的设计周期、更低的建造成本、更低的调试风险等。虚拟调试(Virtual commissioning, VC)^[2]通过仿真技术在虚拟环境中对系统运行逻辑进行测试, 减轻了实机调试的负担, 可有效解决上述问题。

虚拟调试通过执行涉及系统虚拟模型与实际控制器的仿真对系统进行验证^[2], 因此关于虚拟调试的研究多集中在虚拟模型的构建方面, 其中又以逻辑模型方面的研究最为突出。

离散事件系统规范(Discrete event system specification, DEVS)是一种模块化、层次化的形式化机制^[3], 对于复杂逻辑的描述具有较大的优势,

被广泛应用于虚拟调试的逻辑模型构建中。利用 DEVS 构建逻辑模型主要有 3 种形式: 第 1 种是利用 DEVS 的原型构建设备逻辑模型^[4], 主要适用于不考虑设备交互情况下的行为逻辑描述; 第 2 种是将 DEVS 原型作为三阶建模框架(Three-phase modelling framework)中各阶模型的构建方法^[5-6], 通过多种模型的组合描述设备行为逻辑与交互逻辑; 第 3 种是在第 2 种的基础上, 根据可编程逻辑控制器(Programmable logic controller, PLC)的控制机制简化模型组合并扩展 DEVS 原子模型^[7], 其目的是使逻辑模型能够完整地描述实际的控制机制。

综上所述, 目前对于虚拟调试的研究主要集中在逻辑模型的构建方法方面, 而从系统整体上考虑虚拟调

* 基金项目: 四川省科技计划项目(2020YJ0215)。

试的实现,并快速设计系统控制方案则需要进一步的研究。鉴于此,本文提出一种基于可试验数字孪生体(Experimentable digital twin, EDT)^[8]的虚拟调试架构。EDT是数字孪生技术与仿真技术的结合,在运行阶段无需考虑物理系统,通过对其数字孪生体的重复测试与试验验证系统的功能与性能^[9]。所提的架构从设备层、单元层与系统层3层构建FMS的数字孪生体,重点阐述逻辑建模方法与过程,提出基于逻辑模型实现PLC程序快速设计的方法,并进行实例验证。

基于 EDT 的 FMS 虚拟调试架构

如图1所示,基于EDT的FMS虚拟调试架构包括物理系统与虚拟系统两部分。

物理系统由硬件设备与软件系统组成,软件与硬件之间自上而下传递控制信息,实现递阶控制过程。系统层与单元层之间用Kafka作为消息中间件实现生产指令的调用与执行反馈;单元层与设备层之间用

Redis作为实时数据库实现控制变量的写入与设备状态的读取。

虚拟系统由虚拟模型与驱动引擎组成。设备层的模型是对物理设备几何、运动、逻辑3个维度的映射;单元层、系统层的模型是对各层运行控制逻辑的映射,各层模型之间通过逻辑模型自上而下传递控制信息,自下而上传递执行反馈信息。驱动引擎用于支撑虚拟模型的构建与仿真,包括图形引擎(支撑几何模型的构建与实时渲染)与仿真引擎(支撑逻辑模型的构建与仿真)。

虚拟调试的实现分为两部分:(1)利用驱动引擎与各层模型之间的输入、反馈关系实现虚拟系统的仿真运行;(2)将物理系统运行过程中调用的Kafka指令同步到单元层模型,或将Redis中的控制变量同步到设备层模型,利用驱动引擎实现虚拟系统的仿真运行。

综上所述,逻辑模型是实现FMS数字孪生体仿真的核心,因此逻辑建模是实现基于EDT的FMS虚拟调试架构的一项关键技术。此外,虚拟调试的最终要求是设计系统

的控制方案,因此基于逻辑模型实现的PLC程序的设计是另一关键技术。

基于 DEVS 的 FMS 逻辑建模

1 DEVS 原子模型的扩展

由于DEVS原子模型无法完整描述系统运行控制过程,文献[7]将其输出函数扩展后得到一个九元组:

$$FMS = \langle X, Y, S, \delta_{int}, \delta_{ext}, \lambda_a, \lambda_d, \lambda_m, t_a \rangle$$

其中 λ_a 、 λ_d 和 λ_m 是由 λ 扩展后的输出函数,分别表示到达特定状态时的输出、偏离特定状态时的输出和经过特定时间时的输出,且扩展后的模型允许外部转换函数 δ_{ext} 以多个输入事件作为函数输入。与原始原子模型相比,其优点体现在3个方面:(1)具有一对多的输出关系,即在一个状态转移中允许输出事件集 Y 中多个元素的更改,不再局限于一对一、多对一的输出关系。(2)只要发生状态转移,就允许触发输出函数,保证系统中所有的状态变化能被通知。(3)外部转换函数 δ_{ext} 中的输入事件既可以是输入事件集 X 中的某个事件,又可以是多个事件的组合,通过组合结果和当前状态决策函数输出。

上述扩展形式可以描述与状态变化、时间相关的输出,但是实际情况中还存在与频数相关的输出,如传送带上工件到达指定数量时传送带停止,机器人开始抓取工件,因此本文在上述DEVS原子模型扩展形式的基础上将其再次扩展为一个十元组:

$$E_{FMS} = \langle X, Y, S, \delta_{int}, \delta_{ext}, \lambda_a, \lambda_d, \lambda_m, \lambda_c, t_a \rangle$$

再次扩展后的模型增加了一个输出函数 λ_c ,表示事件或状态出现一定次数时的输出,具体描述为:

$$\lambda_c(\text{Flag}, \text{StateOrEvent}, \text{Count}) = \text{Out}$$

其中Flag表示函数是否执行的标志,该函数可能只在特定条件下存在;StateOrEvent表示需记录出现次数的状态或事件;Count表示触发函数时StateOrEvent需出现的次数,取值范围为 $[0, +\infty)$;Out表示函数的输

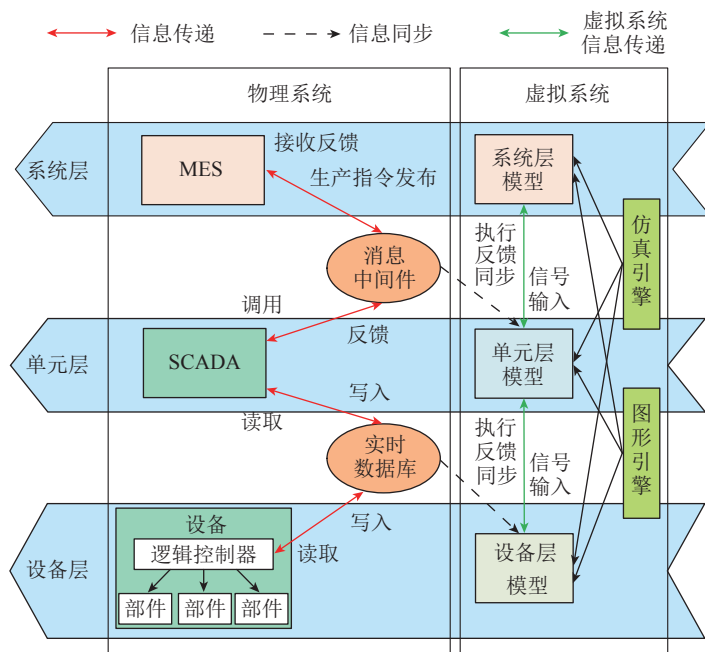


图1 基于 EDT 的虚拟调试架构

Fig.1 Virtual commissioning architecture based on EDT

出结果。文献[7]中的扩展形式虽然定义了输出函数 λ_m ,但是并未给出函数的具体形式,参考 λ_c 的形式可将其描述如下:

$$\lambda_m(\text{Flag}, \text{StateOrEvent}, \text{Count}) = \text{Out}$$

其中 Flag、StateOrEvent、Out 与 λ_c 中含义相同, Time 表示触发函数时 StateOrEvent 需持续的时间,取值范围为 $[0, +\infty)$ 。

文献[7]中的扩展形式允许多个输入事件的组合作为外部转移函数 δ_{ext} 的输入,输入的判断依据是输入事件的值(如“0”、“1”),但是在某些情况中以输入事件的值是否变化作为输入判断依据,如工件在传送带上运行过程中,通过光电传感器时不断产生电脉冲信号(低电平变为高电平或高电平变为低电平),根据电脉冲信号的次数记录工件数量。因此本文扩展了 δ_{ext} 中输入的方式,利用 x^* 描述上述情况,例如“ $x^*=1$ ”表示输入事件 x 的值由“0”变为“1”,“ $x^*=0$ ”表示输入事件 x 的值由“1”变为“0”。

2 FMS 逻辑建模关键流程

FMS 逻辑模型的构建是自底向上的过程,由下层逻辑构建上层逻辑,其流程如图2所示。

本文将 FMS 各层逻辑模型分

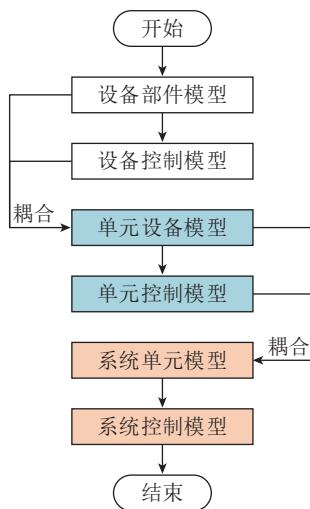


图2 FMS 多层逻辑建模流程

Fig.2 Process of multi-layer logic modeling for FMS

为控制模型与输出模型。控制模型用于描述各类传感信息输入系统时如何决策出控制系统行为的信号;输出模型则描述了控制模型输出的信号与执行器行为的联系。如图2所示,设备层中的控制模型、输出模型分别为设备控制模型与设备部件模型,设备部件模型根据设备中的运动部件构建,设备控制模型则根据设备命令执行过程中部件动作顺序构建,每条设备命令构建一个设备控制模型;单元层的控制模型、输出模型分别为单元控制模型与单元设备模型,单元设备模型由设备控制模型与设备输出模型耦合得到,单元控制模型则与单元指令对应;同理,系统层的控制模型、输出模型分别为系统层控制模型与系统层单元模型,构建方法与单元层的控制模型、输出模型类似。

基于逻辑模型的 PLC 程序设计

PLC 是目前最适用和应用最广泛的工业控制技术^[2], PLC 产品基本上都符合 IEC61131-3 国际标准,该标准中规定了梯形图、功能块图、

语句表、结构文本与顺序功能图 5 种编程语言。梯形图由于表达直观而被广泛使用,其与逻辑模型之间的对应关系如图3所示。

顺序控制设计法是 PLC 编程常用的方法,这种设计方法的本质是用输入量 I 控制代表各步的编程元件(例如存储器位 M),再用它们控制输出量 Q ,步是根据输出量 Q 的状态划分的^[10]。因此基于逻辑模型实现 PLC 程序设计可分为两部分:(1)位地址的映射,将逻辑模型中的各集合映射到 PLC 中的存储单元,并为集合中的元素分配地址;(2)程序段的映射,将逻辑模型状态转移过程中的各元素映射为不同的指令,组成各 PLC 程序段。

1 位地址的映射

S7 系列 PLC 的存储区包括输入映像存储区(I)、输出映像存储区(Q)、位存储器(M)、外部输入寄存器(PI)、外部输出寄存器(PQ)、定时器(T)、计数器(C)、数据块寄存器(DB)与本地数据寄存器(L)^[11],如表1所示,本文主要实现了输入映像存储区、输出映像存储区与位存储器

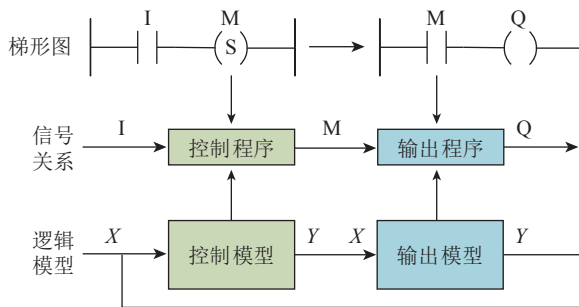


图3 逻辑模型与 PLC 程序的对应关系

Fig.3 Correspondence between logic model and PLC program

表1 PLC 存储区映射规则

Table 1 Mapping rules of storage area for PLC

控制模型	输出模型	PLC 存储区
X_c	—	I
Y_c	X_o	M
S_c	—	M
—	S_o (内部转移前的状态)	Q

的映射。

PLC 程序中控制程序段的输入、输出信号为程序中的输入信号与中间控制信号,而控制模型输入事件集 X_c 、输出事件集 Y_c 表示逻辑模型的输入与中间结果,因此可以分别映射到 PLC 的输入映像存储区与位存储器,如果控制模型的输出事件集 Y_c 与输出模型的输入事件集 X_o 不是完全一对一的关系,则需要对输出模型的输入事件集 X_o 单独映射;控制模型中只存在外部状态转移,状态既作为函数的输入又作为函数的输出,在模型内部变化,因此可将其集合 S_c 映射到 PLC 的位存储器。输出程序段的输出信号是程序的最终输出信号,表示任务执行的启动信号,一般直接通过输入信号(即控制程序段的输出信号)控制,在输出模型中输入事件驱动模型产生外部状态转移,进而由内部状态转移表示任务执行过程,因此可将其状态集 S_o 中表示内部状态转移前的状态映射到 PLC 的输出映像存储区。

2 程序段的映射

控制模型由输入事件触发外部状态转移,进一步由状态转移触发输出函数产生输出事件,根据图 3 中的对应关系可通过控制模型的外部状态转移过程实现控制程序段的设计。如图 4 所示,函数 δ_{ext} 中转移前的状态映射为常开触点指令,输入事件根据表示形式与事件值映射为不同的指令,表示形式“x”根据事件值“0”或“1”映射为常开触点指令或常闭触点指令,表示形式“x*”根据事件值“0”或“1”映射为触点负跳沿检测指令或触点正跳沿检测指令,上述指令作为程序段输入根据组合关系“&”或“||”按顺序串联或并联连接;转移前后的状态分别映射为复位指令与置位指令,转移过程中的输出事件根据事件值“0”或“1”映射为复位指令或置位指令,如果控制模型的输出事件与输出模型的输入事件是

一对多的关系,则需根据控制模型的输出事件将输出模型对应的输入事件映射为相同的指令,上述指令全部作为控制程序段的输出。

输出模型由输入事件触发外部状态转移,进一步由内部状态转移表示执行过程,根据图 3 中的对应关系可通过输出模型的外部状态转移过程实现输出程序段的设计。如图 5 所示,输入事件将事件值“0”或“1”映射为常开触点指令或常闭触点指令,转移后的状态(一般为表示过程开始的状态)映射为输出线圈,上述指令分别作为程序段的输入与输出。

输出函数 λ_m 和 λ_c 分别可映射为 PLC 中的定时器指令与计数器指令。如图 6 (a) 所示,输出函数 λ_m 的输入 StateOrEvent 映射为常开触点指令连接到定时器的置位信号端 (S), Time 映射为定时器时间值输入端 (PV) 的时间值,输出结果 Out 映

射为输出线圈连接到定时器输出值端 (Q)。

如图 6 (b) 所示,输出函数 λ_c 的输入 StateOrEvent 映射为常开触点指令连接到减计数器输入端 (CD), Count 映射为计数器初值输入端 (PV) 的计数初值,输出结果 Out 映射为输出线圈连接到计数器状态输出端 (Q)。

实例验证

图 7 所示为基于本文所提架构的某 FMS 虚拟调试实现与验证框架。该 FMS 由 3 个加工单元、1 个仓储单元与 1 个物流单元组成,加工单元包括雕刻设备(雕刻机)与上下料设备(气动夹臂、工业机器人),仓储单元包括堆垛机与立体仓库,物流单元主要指传送带,由滚筒线、出货台、入货台、转角机等组成。

如图 7 底部所示,物理系统由硬件设备与软件系统(如 MES、

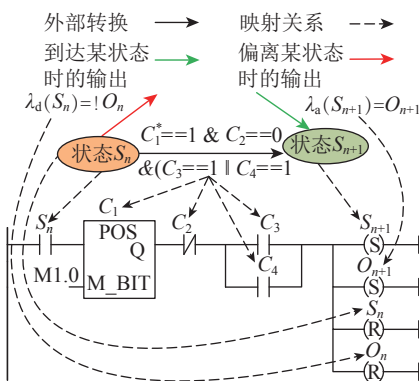


图 4 控制程序段的映射
Fig.4 Mapping for control program segment

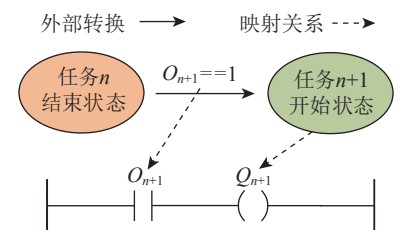
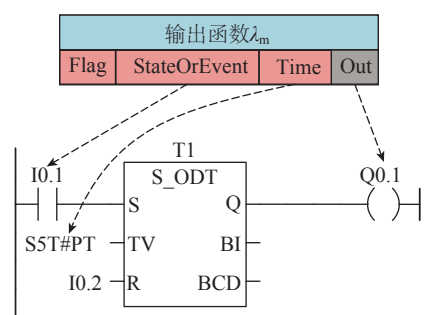
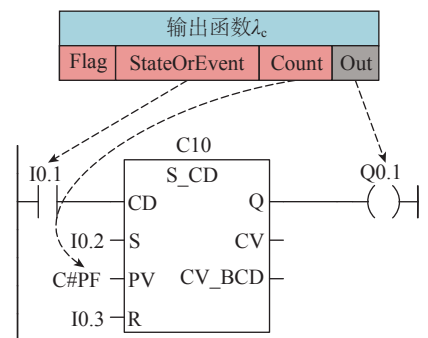


图 5 输出程序段的映射
Fig.5 Mapping for output program segment



(a) 定时器指令映射



(b) 计数器指令映射

图 6 特殊指令的映射

Fig.6 Mapping for special instruction

SCADA等)组成。在物理系统运行过程中, MES通过Kafka将生产指令发送至SCADA, SCADA对生产指令进行解析,将控制变量写入Redis。设备根据Redis中的变量值触发相应的控制程序,驱动设备执行相应的动作。

如图7顶部所示,虚拟系统通过对上述控制信息传递过程的映射,自底向上逐层构建设备、单元、系统的逻辑模型(包括控制模型与输出模型)。其中,设备逻辑模型作为单元输出模型参与单元逻辑模型的构建,单元逻辑模型作为系统输出模型参与系统逻辑模型的构建。在虚拟系统运行过程中,根据各层逻辑模型中控制模型与输出模型的耦合关系自上而下运行模型,再通过设备逻辑模型驱动设备三维模型运行;通过观

察、分析三维模型运行情况,反复修改、验证各层逻辑模型。最后,利用所提的映射规则,根据逻辑模型设计PLC程序,利用编程软件编写、调试后下发到物理系统,完成最终的调试验证。

基于EDT的虚拟调试包括以下两种方式:一种是在纯虚拟环境下,针对某一层的逻辑模型,模拟模型的输入,再通过仿真引擎驱动逻辑模型,实现虚拟系统的运行;另一种是在半实物环境下,结合物理系统的运行过程,通过Kafka与Redis提供的API在虚拟系统中调用生产指令或读取控制变量,将生产指令或控制变量解析处理后,分别作为单元逻辑模型与设备逻辑模型的输入驱动虚拟系统运行,实现对系统不同层次控制方案的调试验证。

虚拟系统与物理系统实现逐层控制的本质是相同的,都是自上而下不断地分解输出,物理系统的输出通过解析处理后可用于控制虚拟系统,反之同样可行。虚拟系统系统层和单元层产生的输出可利用API分别传入Kafka与Redis中,物理系统从中可调用该输出,对其解析处理后可用于物理系统的运行控制,对于实现以虚控实以及基于数字孪生的车间装备智能控制具有一定的参考意义。

为进一步验证所提架构中的关键技术,结合图7中的FMS对其进行描述,图8所示为以气动夹臂为例构建的设备逻辑模型,图9所示为加工单元逻辑模型。

设备逻辑模型中的输出模型根据设备中运动部件构建,通过部件运

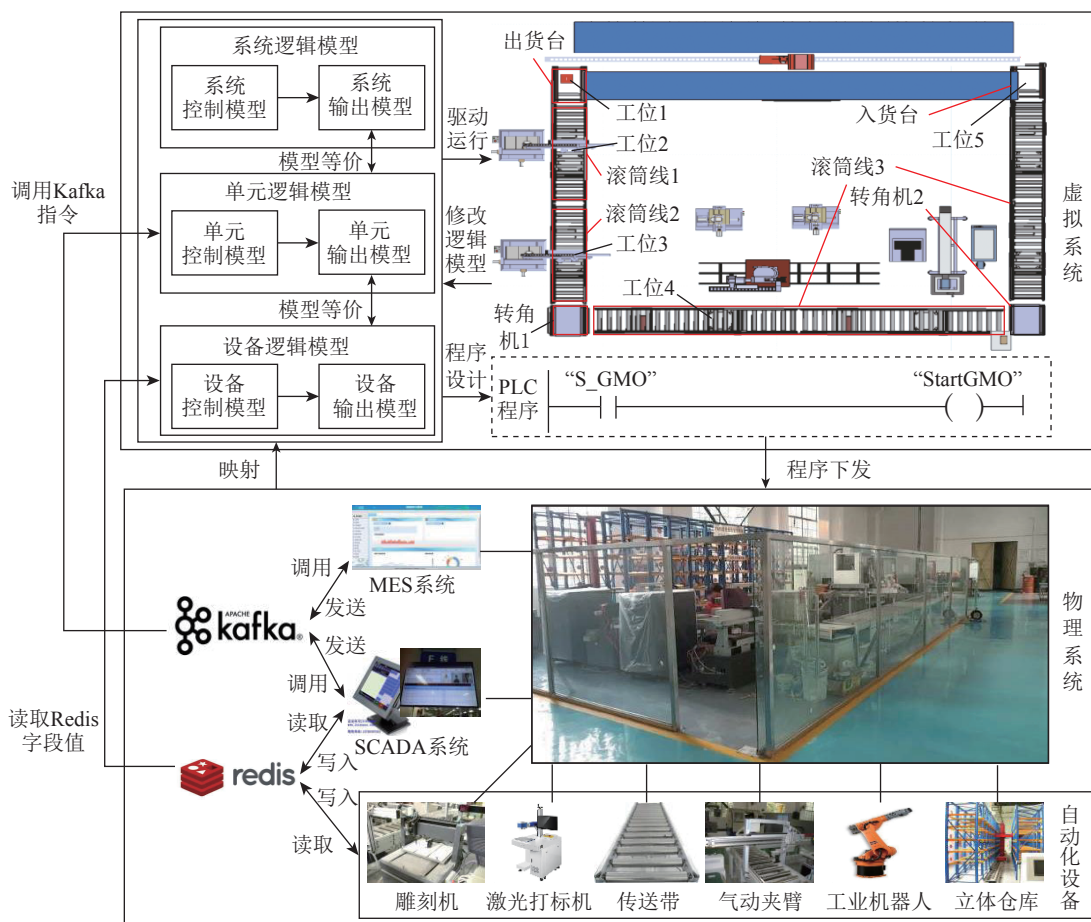


图7 某FMS的虚拟调试实现与验证框架

Fig.7 Implementation and verification framework of virtual commissioning for FMS

动确定模型的输入输出,控制模型则根据设备需执行的命令构建,不同的命令用不同的控制模型表示,图8

(d)所示为其中一个控制模型。单元逻辑模型的输出模型根据单元中的设备构建,如图9所示的气动夹臂

与雕刻机可分别构建输出模型,该模型可通过对设备逻辑模型的简化等价来构建,图9(a)所示的模型即为

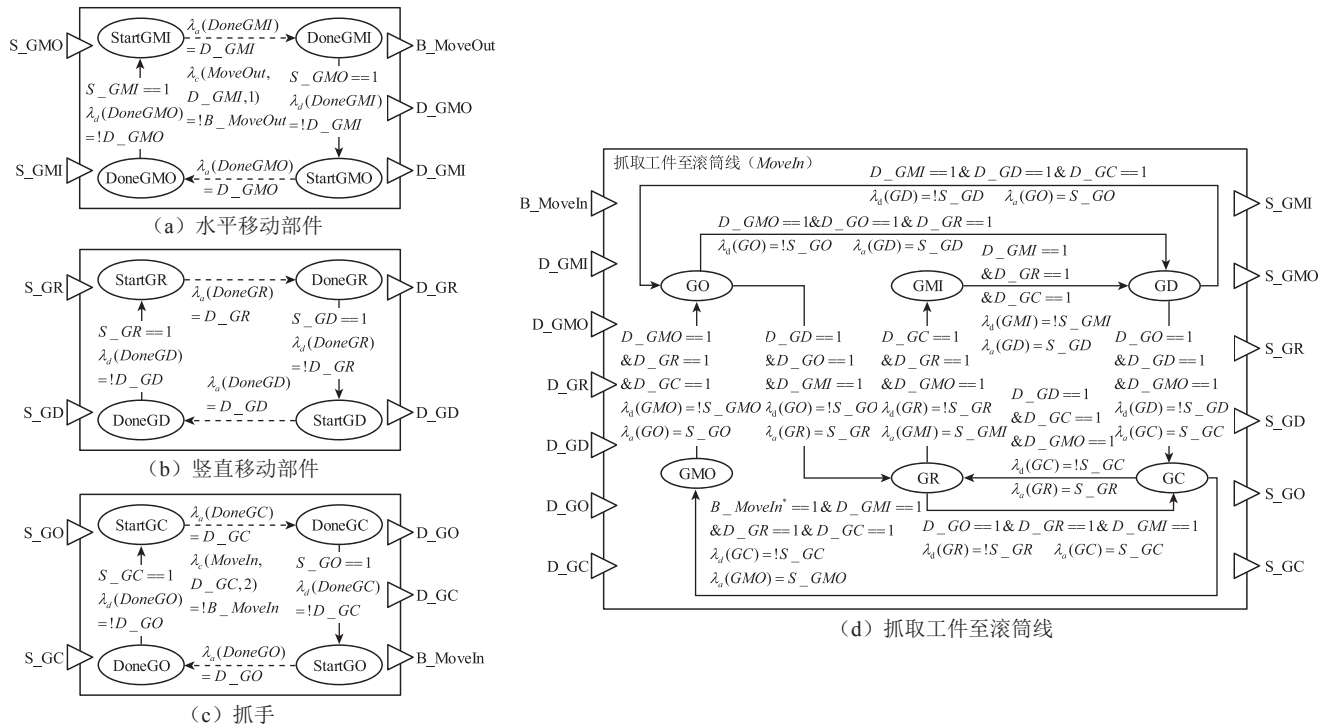


图8 设备逻辑模型
Fig.8 Logic model of device

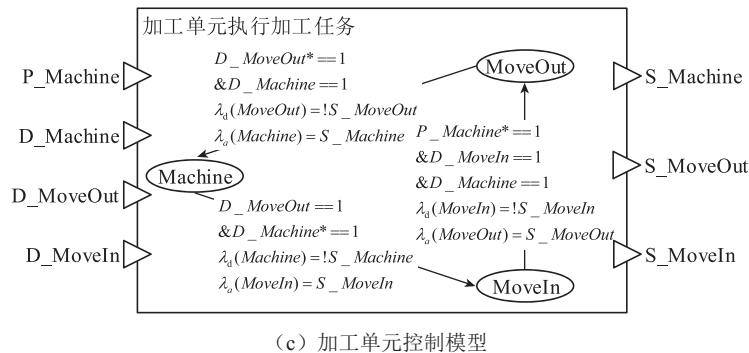
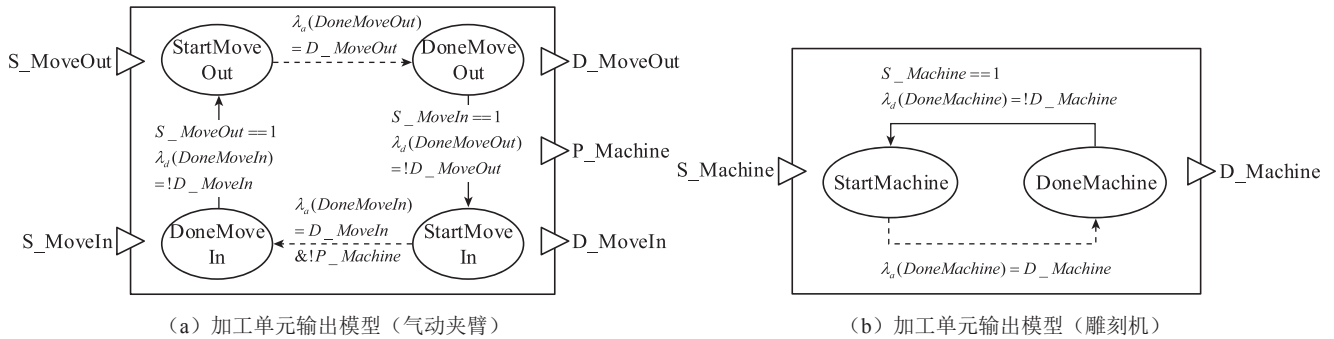


图9 加工单元逻辑模型
Fig.9 Logic model of manufacturing unit

图 8 所示模型的等价模型。

在本文的案例中,仓储单元与加工单元的运行过程都是连续的,即单元开始执行任务时各设备按顺序依次执行设备命令,而物流单元的运行需要与仓储单元、加工单元不断交互,其过程是不连续的,因此其逻辑模型有所不同,如图 10 所示。

物流单元由于需要与仓储单元、加工单元不断交互,因此其控制模型的输入需要考虑其他单元的运行结果,如图 10 (c) 中的 DL_Transport1 表示仓储单元执行结束时的输出, DL_Transport2、DL_Transport3 与 DL_Transport4 则分别表示 3 个加工单元执行结束时的输出。此外,物流单元的运行通过出货台、入货台以及不同的转角机与滚筒线的联动实现,因此控制模型的输出可同时作为不

同输出模型的输入,即存在一对多的耦合关系,如图 10 (c) 中的彩色箭头所示。

逻辑模型可以清晰地描述系统的控制决策与执行过程,是实现系统基于状态的虚实同步以及以虚控实的重要基础。逻辑模型的构建与验证通过仿真引擎实现,本文采用 DEVS++ 程序库^[12]作为该仿真引擎,通过对其二次开发实现本文提出的 DEVS 扩展形式。DEVS++ 是以 C++ 语言编写的开源程序库,库中只包含头文件与源文件,因此可直接通过编程软件对其进行二次开发,本文采用 Visual Studio 2012 作为二次开发工具。

二次开发的内容分为两部分:一是实现模型形式的扩展,通过改写 Atomic 类中的函数 delta_x(), 允许外

部转移函数输入可为多个输入事件组合的情况,以及以输入事件值的变化作为输入的情况,新增函数 delta_y_arrival()、delta_y_departure()、delta_y_timed() 与 delta_y_counted() 实现多种输出;二是实现模型的构建,通过继承 Atomic 类并在派生类中重写上述函数以实现模型功能。二次开发的流程如图 11 所示,根据实例中的模型设计开发调试界面,其运行结果如图 12 所示。

限于篇幅,本文以图 8 所示的逻辑模型为例演示基于逻辑模型的 PLC 程序设计过程,首先利用表 1 中的存储区映射规则实现关键位地址的映射,建立符号表。然后利用本文提出的程序段映射规则设计气动夹臂的控制程序段与输出程序段,将程序段按顺序排列构建完整的 PLC

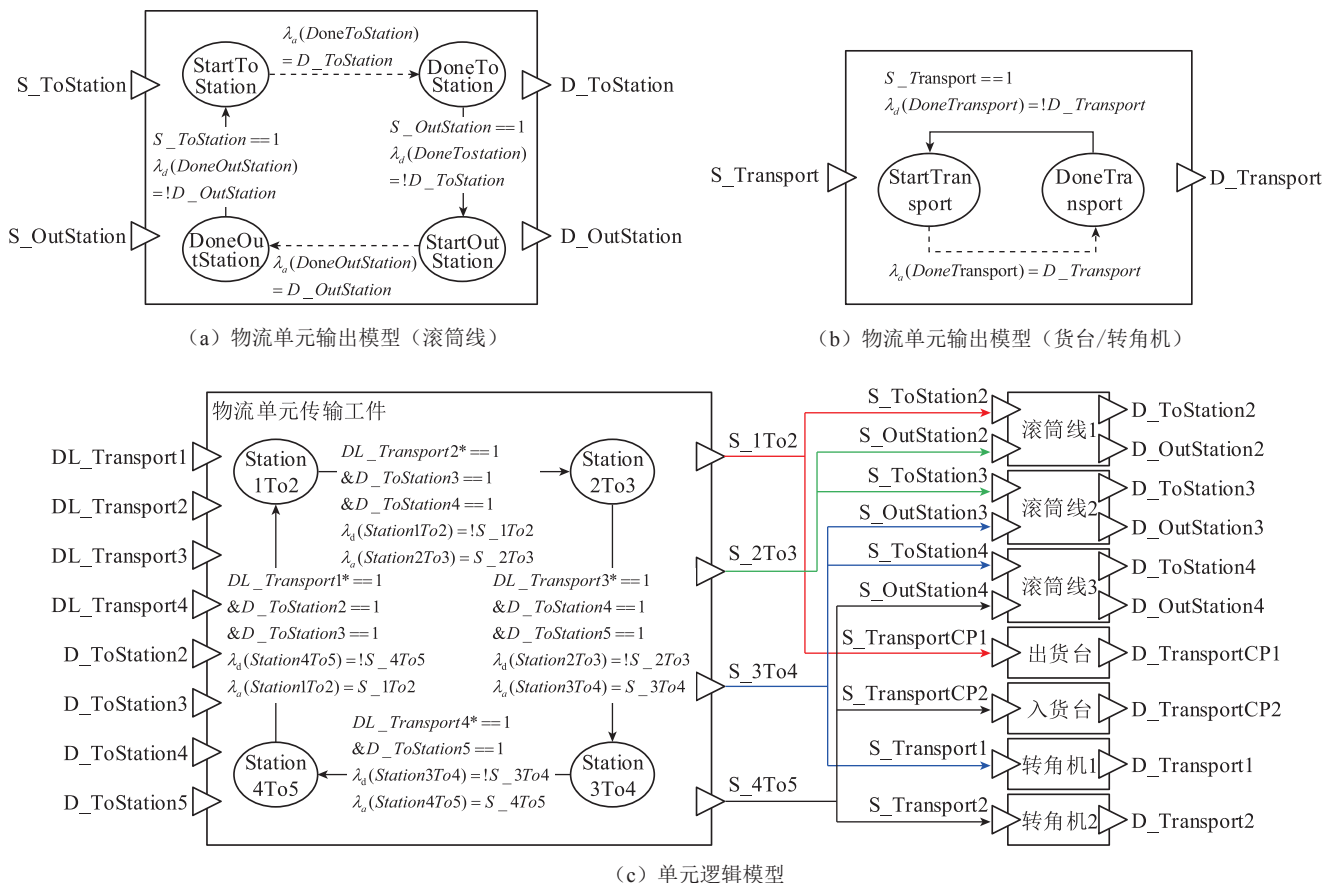


图 10 物流单元逻辑模型

Fig.10 Logic model of logistic unit

程序。通过 STEP7 编程软件编写 PLC 程序, 并利用 S7-PLCSIM 仿真, 结果证明达到了预期的控制效果, 如图 13 (a) 所示, PLC 代码如图 13 (b) 所示。逻辑模型相比 PLC 程序具有规范性强、可重用性高等特点, 其构建过程消耗的时间较短, 修改调整过程简单, 因此通过映射规则设计 PLC 程序可比直接设计节省大量时间。

结论

针对缺乏从系统整体上考虑虚拟调试的实现以及实现系统控制方案设计等问题, 本文提出一种基于 EDT 的 FMS 虚拟调试架构, 根据其两项关键技术主要做了以下工作:

(1) 在 DEVS 扩展形式的基础上对其原子模型的输出函数再次进行扩展、补充, 使其对系统运行控制过程的描述更加全面。

(2) 基于扩展后的逻辑模型, 提出逻辑模型到 PLC 程序的映射规则, 实现了位逻辑运算指令、计数器指令与定时器指令等常用指令的映射, 为快速、准确地设计 PLC 程序提供了一种切实可行的方法。

本文建立了逻辑模型到 PLC 程序常用指令的映射规则, 但 PLC

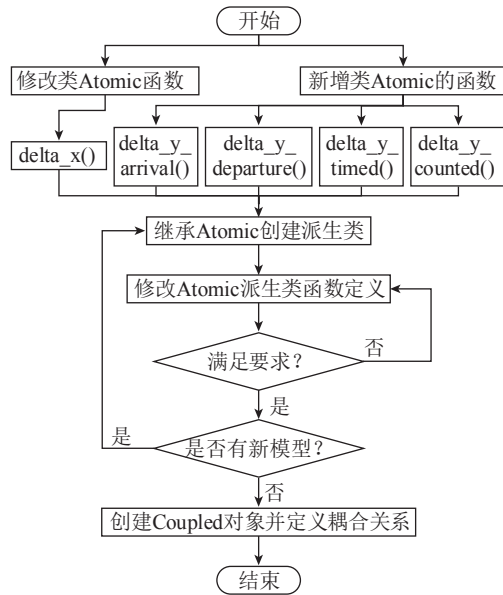


图 11 DEVS++ 二次开发流程

Fig.11 Second development process of DEVS++

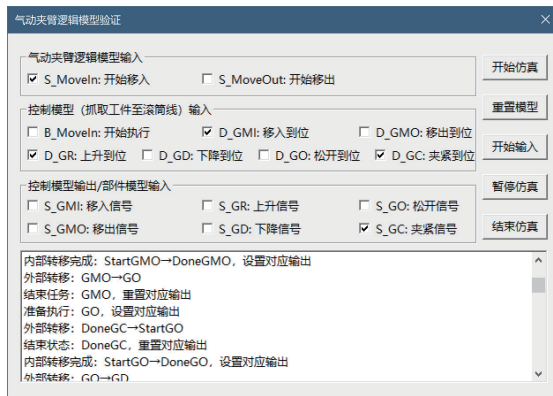


图 12 逻辑模型仿真验证

Fig.12 Simulation and verification of logic models

(a) PLC程序运行结果

(b) PLC代码

图 13 部分 PLC 程序及其运行

Fig.13 Part of PLC program and its operation

程序指令繁多,完全实现逻辑模型到 PLC 程序的映射需进一步的扩展。此外,建立逻辑模型到 PLC 程序的自动生成算法也是后续研究的重点。

参考文献

- [1] 张根保. 自动化制造系统[M]. 北京:机械工业出版社,2017:14-18.
- ZHANG Genbao. Automatic manufacturing system[M]. Beijing: China Machine Press, 2017: 14-18.
- [2] LEE C G, PARK S C. Survey on the virtual commissioning of manufacturing systems[J]. Journal of Computational Design and Engineering, 2014, 1(3): 213-222.
- [3] 邱晓刚, 段伟. DEVS 研究进展及其对建模与仿真学科建立的作用[J]. 系统仿真学报, 2009, 21(21): 6697-6704, 6709.
- QIU Xiaogang, DUAN Wei. Research development of DEVS and its function to modeling and simulation discipline[J]. Journal of System Simulation, 2009, 21(21): 6697-6704, 6709.
- [4] LEE J Y, LEE K, PARK S. Virtual commissioning for an Overhead Hoist Transporter in a semiconductor FAB[J]. International Journal of Production Research, 2020, 58(22): 6890-6898.
- [5] PARK S C. A methodology for creating a virtual model for a flexible manufacturing system[J]. Computers in Industry, 2005, 56(7): 734-746.
- [6] PARK H T, SEONG K Y, DANGOL S, et al. An approach to obtain a PLC program from a DEVS model[C]//International Conference on Icinco, Funchal: DBLP, 2008: 87-93.
- [7] KO M, PARK S C, CHOI J J, et al. New modelling formalism for control programs of flexible manufacturing systems[J]. International Journal of Production Research, 2013, 51(6): 1668-1679.
- [8] MICHAL S, MARC P, LINUS A. Experimentable digital twins: Streamlining simulation-based systems engineering for industry 4.0[J]. IEEE Transactions on Industrial Informatics, 2018, 14(4): 1722-1731.
- [9] 李浩, 李兵, 王昊琪, 等. 一种智能产线可试验数字孪生体建模方法: CN111046589B[P]. 2020-11-13.
- LI Hao, LI Bing, WANG Haoqi, et al. A modeling method for a testable digital twin of an intelligent production line: CN111046589B[P]. 2020-11-13.
- [10] 廖常初. S7-300/400 PLC 应用技术[M]. 北京:机械工业出版社,2010.
- LIAO Changchu. Application technology for S7-300/400 PLC[M]. Beijing: China Machine Press, 2010.
- [11] 陈章平, 杨泽, 沈国宇, 等. 西门子 S7-300/400 PLC 控制系统设计与应用[M]. 北京:清华大学出版社,2009.
- CHEN Zhangping, YANG Ze, SHEN Guoyu, et al. Design and application of Siemens S7-300/400 PLC control system[M]. Beijing: Tsinghua University Press, 2009.
- [12] 燕美玲, 张学军. DEVS++ 程序库及应用[J]. 电气自动化, 2018, 40(6): 4-6, 97.
- YAN Meiling, ZHANG Xuejun. DEVS++ program library and its applications[J]. Electrical Automation, 2018, 40(6): 4-6, 97.

通讯作者: 丁国富, 教授、博士生导师, 研究方向为数字化设计与制造、智能制造、数字孪生, E-mail: dingguofu@swjtu.edu.cn.

Logic Modeling and PLC Program Design based on DEVS for Flexible Manufacturing System

WANG Bing, JIANG Haifan, DING Guofu

(School of Mechanical Engineering, Southwest Jiaotong University, Chengdu 610031, China)

[ABSTRACT] Aiming at problems of complicated logic, long commissioning cycle and high commissioning cost in the design process of flexible manufacturing system control scheme, a virtual commissioning method based on experimentable digital twin was proposed. This method established the twin model of physical system in a layered manner, and established a three-layer logic model of equipment layer, unit layer, and system layer based on the extended form of discrete event system specification. The logic model was repeatedly verified and modified through simulation. The mapping rule from logic model to PLC program was established to realize the rapid design of PLC program. Finally, taking a FMS as an example, the process of logic modeling and PLC program design were demonstrated in detail. The feasibility and effectiveness of the proposed method were verified.

Keywords: Flexible manufacturing system; Experimentable digital twin; Virtual commissioning; Logic modeling; Discrete event system specification (DEVS); Programmable logic controller

(责编 一元)