

面向智能制造加工的虚拟调试系统开发与应用

禹鑫燚, 陆文祥, 柏继华, 王正安, 欧林林

(浙江工业大学 信息工程学院, 杭州 310023)

摘要: 为了解决智能制造加工系统的调试成本高、周期长及风险大等问题, 研究了面向智能制造加工的虚拟调试技术; 以智能制造加工系统中机器人、PLC、数控车床及加工中心为研究对象, 首先通过工业以太网进行组网并与 PC 机进行通讯, 基于 OSGI. NET 插件框架开发了交互控制软件, 将每个设备驱动做成插件, 使得每个插件能够实现当前设备的数据采集、监控及控制; 并且在 RoboDK 软件中建立了系统虚拟的 3D 模型, 通过 Redis 数据库实现了控制设备和虚拟模型之间的信号交互; 然后使用了 PLC、机器人控制器、数控车床及加工中心控制器对系统 3D 模型进行虚拟调试, 最后通过实验验证了加工系统的有效性。

关键词: PLC; 机器人; OSGI. NET; RoboDK; 虚拟调试

Development and Application of Virtual Commissioning System for Intelligent Manufacturing Processing

Yu Xinyi, Lu Wenxiang, Bai Jihua, Wang Zhengnan, Ou Linlin

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract: Problems of high cost, long term and high risk of debugging generally exist in intelligent manufacturing processing system. In order to solve these problems, a virtual commissioning technology for intelligent manufacturing processing is investigated. Taking the robot, PLC, CNC lathe and machining center in the intelligent manufacturing processing system as the research object, an Ethernet network is firstly set up to enable the communication with PC. Then, an interactive control software is developed based on the OSGI. NET plug-in framework, in which each device driver is made into a plug-in, which can realize the data acquisition, monitoring and control of the current device. Moreover, a virtual 3D model of the system is established in RoboDK software, then the signal interaction between the control device and the virtual model is realized through the Redis database. Furthermore, the PLC, robot controller, CNC lathe and machining center controller were used to debug the 3D model of the system. Finally, the effectiveness of the processing system is verified by an experiment.

Keywords: PLC; robot; OSGI. NET; RoboDK; virtual commissioning

0 引言

21 世纪以来, 物联网、云计算、大数据、移动通信、人工智能等新技术在制造业广泛应用, 制造系统集成式创新不断发展, 形成新一轮工业革命的重要驱动力。面对新一轮工业革命, 世界各国都在出台制造业转型战略, 美国提出“先进制造业伙伴计划”、德国提出“工业 4.0 战略计划”, 都将智能制造作为本国构建制造业竞争优势的关键举措^[1-2]。与此同时, 在“制造强国”和“网络强国”大战略背景下, 我国也先后出台了“中国制造 2025”和“互联网+”等制造业国家发展实施战略, 其核心是促进新一代信息技术与制造业深度融合, 大力发展智能制造^[3-4]。数字孪生 (Digital Twin)^[5-7], 作为实现物理世界与信息世界交互与融合的有效方法, 是指通过数字技术来复制物理对象,

模拟对象在现实环境中的行为, 对整个工厂的生产过程进行虚拟仿真, 从而提高制造企业产品研发、制造的生产效率。数字孪生正在成为智能制造新趋势, 已经在 2017、2018 和 2019 年连续三年被世界领先的科技研究集团 Gartner 评为年度十大战略性技术, 预测在不久的将来将有数十亿台设备拥有数字双胞胎, 并且到 2020 年, 估计将有 210 亿个连接的传感器和端点投入使用。

虚拟调试 (Virtual Commissioning)^[8-10] 是数字孪生最好的应用, 虚拟调试技术是在虚拟环境中调试控制设备的代码, 然后通过虚拟仿真来测试和验证系统方案的可行性, 再将调试代码应用到真实的场景中。该虚实融合的虚拟调试技术的优势在于降低了调试成本和风险、减少了现场调试时间、提升了工作质量、验证 PLC 的逻辑及产品的可行性。国内外有很多学者对虚拟调试技术进行了研究, Koo 提出了虚拟工厂模型框架, 并应用于汽车装配线中, 通过建模和仿真验证了 PLC 逻辑^[11]。刘玉玲在 Roboguide 平台对焊接机器人增材制造进行了虚拟仿真, 有效缩短了机器人焊接时间, 提高了机器人工作效率^[12]。黄永飞开发了一种面向三维虚拟生产线的 PLC 仿真控制系统, 实现了基于 PLC Editor 的虚拟自动生产线的仿真控制^[13]。杨大宇基于

收稿日期:2019-06-04; 修回日期:2019-06-28。

基金项目:国家重点研究发展计划“智能机器人”重点项目(2018YFB1308000);机器人技术与系统国家重点实验室开放基金项目(SKLR2013MS06)。

作者简介:禹鑫燚(1979-),男,浙江省余姚市人,博士,讲师,主要从事嵌入式、工业机器人控制方向的研究。

Autoform 软件实现了对汽车覆盖件拉伸模具的参数化虚拟调试, 使模具在制造前就消除了成形缺陷^[14]。王春晓采用 Modelica 进行数控机床多领域建模并虚拟调试, 对机床实机调试具有指导意义^[15]。

本文首先设计了面向智能制造加工的虚拟调试系统的总体方案, 系统包括三部分: 物理控制设备、交互控制软件和虚拟 3D 模型; 其次根据系统的功能需求开发了 PC 端的交互控制软件, 实现了物理与虚拟信号的交互融合; 然后在 RoboDK 仿真软件中建立了机器人、立体仓库、数控车床加工中心等 3D 模型, 通过 PLC、机器人控制器、车床控制器及加工中心控制器进行虚拟调试, 调试过程中发现了 PLC 程序设计的不合理之处并进行了改善; 最后搭建了实验平台, 通过实验对智能制造加工系统的可行性进行了验证。

1 系统总体方案设计

整个系统由物理控制设备、交互控制软件和虚拟 3D 模型三大部分组成。物理控制设备以 PLC 为中央控制器与机器人控制器、车床控制器、加工中心控制器接入同一交换机下, 通过工业以太网进行通信。PC 端设计一款交互控制软件, 采集、监控物理设备的数据, 同时实现系统中物理信号与虚拟信号的交互。在 RoboDK 软件中建立加工系统中机器人、立体仓库、数控车床及加工中心等 3D 模型, 用来模拟真实的加工系统。系统总体方案如图 1 所示。

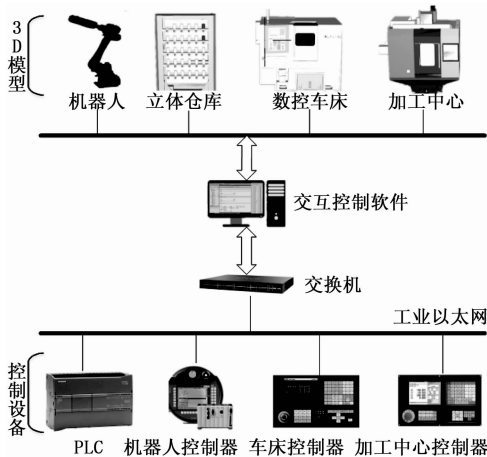


图 1 系统总体方案图

以中央控制器 PLC 为核心, 对整个系统进行逻辑控制, 并通过交互控制软件, 进行物理信号与虚拟信号的匹配, 实现以下功能: 1) 实现与机器人模型运动指令、运行状态及数据的交互; 2) 实现数控车床模型安全门开关、卡盘松紧及加工工艺的执行; 3) 实现加工中心模型安全门开关、卡盘松紧及加工工艺的执行; 4) 实现与立体仓库模型中工件状态信息的交互。信号匹配完成后对加工系统进行虚拟调试来完成加工任务, 加工任务由以下几部分组成: 1) 机器人夹取立体仓库中毛坯工件进行工件出库; 2) 机器人往数控车床上料; 3) 数控车床加工; 4) 机器人夹取加工完

成的半成品往加工中心上料; 5) 加工中心加工; 6) 机器人夹取成品入库。

2 交互控制软件设计

交互控制软件在 Visual Studio 2015 平台上进行开发, 基于 OSGI. NET 插件框架进行软件设计, 数据层采用 C# 编程语言编写, 显示层采用 WinForm 高效而又强大的控件集设计。OSGi. NET 插件框架, 是 OSGi R4.2 规范移植到 .NET 平台的实现。具有面向服务架构、动态模块化和模块扩展三大功能。支持控制台、WinForm、WPF、ASP. NET 和移动平台等任意 .NET 应用环境, 其体系结构如图 2 所示。

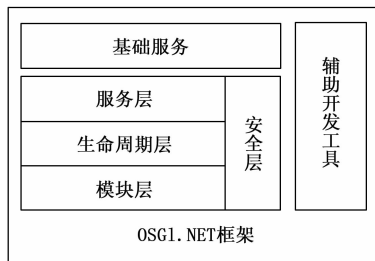


图 2 OSGI. NET 体系结构图

交互控制软件的总体设计框架如图 3 所示。根据 OSGI. NET 插件框架的模块化和插件化的功能, 将每个设备驱动做成插件, 每个插件被独立开发, 设计了相应的采集层、显示层和数据层。采集层: 通过相应设备的应用编程接口 (API), 做成数据采集器, 实时采集设备的数据; 显示层: 实时显示物理信号和虚拟信号的详细信息; 数据层: 将物理信号存储到 Redis 数据库, 同时读取 Redis 数据库中的虚拟信号写入物理设备。使用 Redis 高性能数据库, 将物理信号和虚拟信号进行缓存, 保证了信号交互的实时性。使用 Python 脚本程序读取 Redis 数据库中物理信号来驱动 RoboDK 中 3D 模型, 同时将模型的虚拟信号存储到 Redis 数据库中, 最终反馈到物理控制设备中。

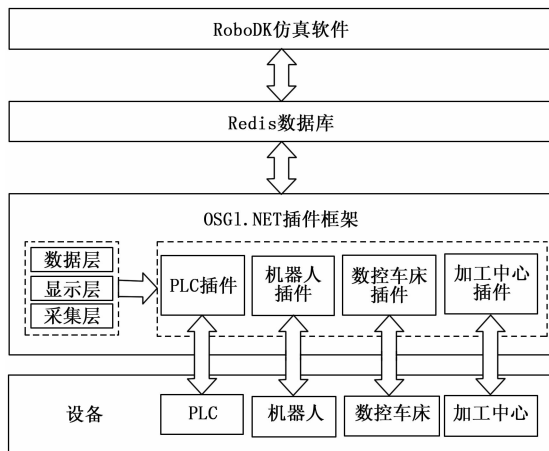


图 3 交互控制软件总体设计框架图

交互控制软件由系统插件和设备插件组成。其中系统

插件用于管理设备插件的生命周期，负责启动、停止及卸载设备插件，能够挑选任意的插件组装进行虚拟调试。机器人插件能够实现与目标机器人控制进行连接、对关节角及对输出信号读取。PLC 插件能够与目标 PLC 进行连接，对 IO 信号进行读写。数控车床插件能够与数控车床控制器进行连接，对 IO 信号进行读写，省去了数控车床与 PLC 的硬件 IO 接线。加工中心插件能够与加工中心控制器进行连接，对 IO 信号进行读写，省去了加工中心与 PLC 的硬件 IO 接线。具体功能如图 4 所示。

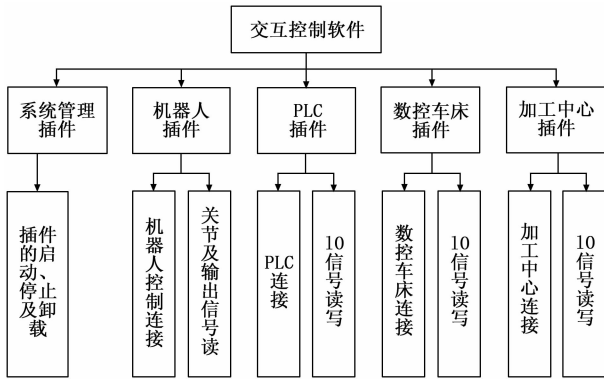


图 4 交互控制软件功能图

软件中每一个设备驱动都由独立的插件所生成，拥有自己的业务逻辑及交互界面。插件具有动态性和隔离性，能够被动态安装、启动、停止及卸载，同时有独立的类型加载器和类型空间。OSGi. NET 插件框架启动时，通过 BundleRuntime 即插件运行时，它是插件的运行容器，会从各个插件目录中逐一加载和启动所有插件。

以开发 PLC 插件为例，基于 OSGI. NET 插件框架在 winform 下开发插件步骤如下：1) 首先在 Visual Studio 2015 中创建一个 winform 主应用程序，作为启动 OSGI 运行时环境的入口程序；2) 添加核心的 UIshell. OSGI 引用；3) 在 Program. cs 文件的程序入口点 Main 函数中创建启动插件运行时 BundleRuntime，从插件框架获取 PageFlowService 服务，利用该服务获取主界面，然后创建该界面实例，并运行；4) 在主程序的输出目录下创建一个 Plugins 文件夹，在该文件夹下建立一个 winform 工程项目，这个项目即为 PLC 插件，BundleRuntime 会自动从 Plugins 目录下加载并启动插件。插件由一个清单文件 (Manifest. xml)、插件本地程序集、插件所需资源和其他文件组成。清单文件 Manifest. xml 可以定义插件的基本信息、插件的激活状态信息、插件类加载的运行时信息及插件的扩展定义信息；5) PLC 插件的界面设计采用两个 listView 控件来进行 PLC 的 IO 信号的更新显示，每行信号包括信号名、虚拟键名、输入/输出地址及当前值属性，信号名为当前信号的描述，虚拟键名为当前信号在 Redis 中的键名，输入/输出地址为 PLC 的输入/输出地址，当前值为输入/输出地址中的信号值，该值会根据键名存储到 Redis 中对应的键值中。在 listView 的输入输出列表范围内进行右击，能够选

择添加信号、修改信号、删除信号、强制信号及取消强制等功能，信号的信息会写入到. ini 配置文件中，PLC 插件操作界面如图 5 所示；6) PLC 插件的后台创建了 3 个线程设计来提高软件的执行速度和运行效率，分别为读 PLC 线程、写 PLC 线程、刷新界面线程。读 PLC 线程的任务是负责连接目标 PLC，实时读取 PLC 的输出信号并存储到 Redis 数据库中。写 PLC 线程的任务是负责连接目标 PLC，从 Redis 数据库中实时获取设置中所需的信号值并写入到 PLC 对应的输入地址中。刷新界面线程的任务是从 Redis 数据库中获取设置信号值并实时显示在界面上，便于观察信号的变化。



图 5 PLC 插件操作界面

开发完成之后运行插件，PLC 插件具体的运行流程图如图 6 所示。首先设置目标 PLC 的 IP 地址，界面上会加载配置文件中已有的信号信息，还需添加信号可以在右击菜单中进行逐行添加，信号添加完成之后运行插件，成功连接 PLC 之后将启动 3 个线程。

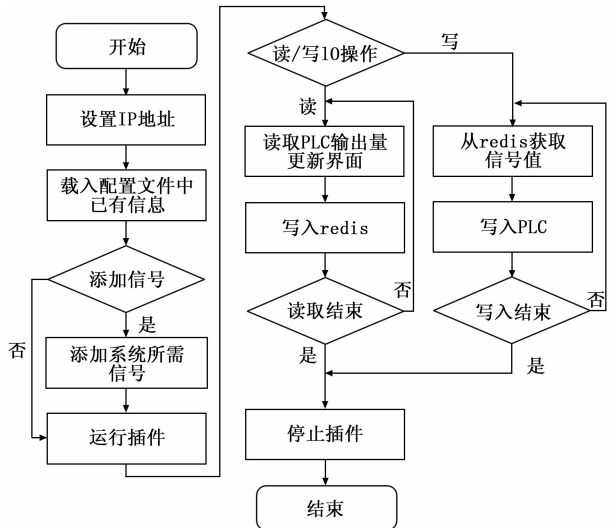


图 6 PLC 插件运行流程图

基于 PLC 插件的开发，同样实现了机器人、数控车床及加工中心插件，数控车床和加工中心插件开发过程与

PLC 插件类似, 机器人插件开发中只对机器人关节角数据及输出信号进行读取, 显示在界面上的同时存储到 Redis 数据库。系统管理插件在界面上使用 DataGridView 控件将插件目录中的插件列出, 并且使用了 3 个按钮, 分别实现设备插件的启动、停止和卸载。系统管理与其他设备插件操作界面如图 7 所示。



图 7 系统管理及其他插件操作界面

3 系统的虚拟调试

虚拟调试可以在项目开始的早期, 利用物理控制设备和虚拟的 3D 模型来模拟整个加工过程, 不仅减少了调试的时间和成本, 而且在办公室就能对控制设备程序进行逻辑检查, 最终验证产品设计的可行性。在 RoboDK 软件上建立加工系统的模型, 然后通过交互控制软件来调试该模型。RoboDK 软件中加工系统工作站如图 8 所示。

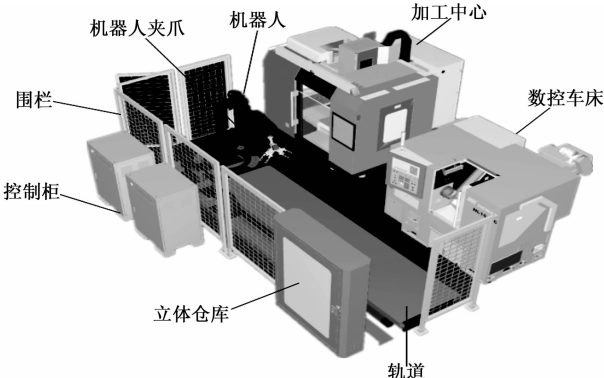


图 8 加工系统工作站示意图

加工系统工作站由以下几部分组成: 汇博 20 kg 型机器人、机器人夹爪、Neway 数控车床、Neway 加工中心、立体仓库、控制柜、机器人轨道及围栏。立体仓库中每个仓位存放着工件, 机器人负责搬运工件进行出库、上料及入

库, 立体库中使用传感器来模拟 RFID, 记录工件的状态, 数控车床负责加工毛坯工件, 加工中心负责加工半成品。

中央控制器 PLC 的控制程序用梯形图语言编写, 机器人控制器的执行程序由汇博机器人脚本语言编写, 在数控车床和加工中心的控制主板中使用 G 代码编程。

加工系统的虚拟调试环境搭建完成后, 通过交互控制软件实现物理控制设备与工作站中模型的信号交互, 与每个设备程序中相对应的输入输出信号如图 9 所示。

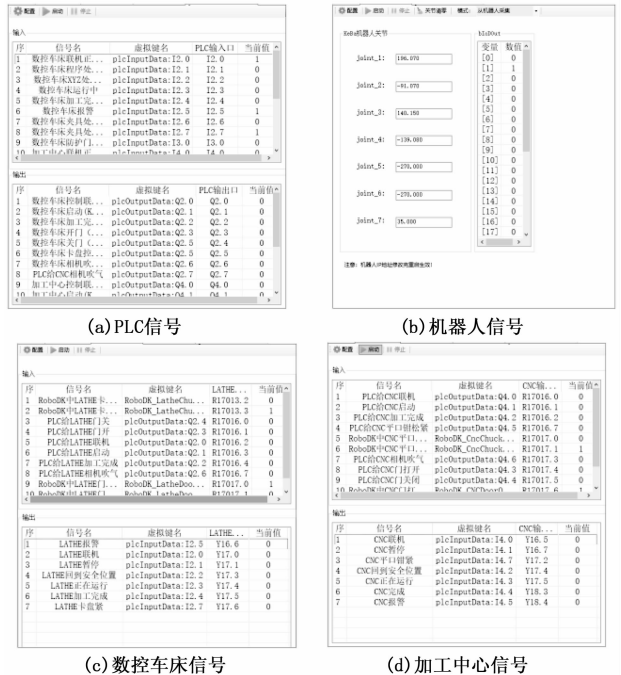


图 9 设备输入输出信号图

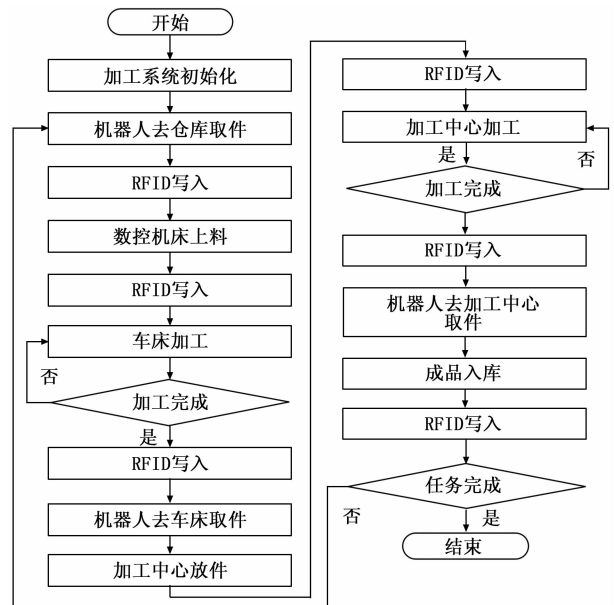


图 10 工作站运行流程图

加工系统工作站运行流程图如图 10 所示。程序启动, 首先是系统的初始化, 机器人回到初始位置, 数控车床和

加工中心的卡盘紧和安全门关闭。初始化完成之后 PLC 根据指令调度机器人去立体仓库的相应仓位取毛坯工件，取件时候将工件的状态写入仓位中的传感器，即 RFID 写入，然后 PLC 调度机器人夹取毛坯件往数控车床上料，完成后进行 RFID 写入，更新工件状态，PLC 请求执行相应的数控车床加工程序，数控车床开始对毛坯工件加工，加工完成之后，再进行 RFID 写入，更新工件状态，PLC 调度机器人去数控车床取件，然后夹取半成品放置到加工中心，完成后进行 RFID 写入，更新工件状态，PLC 请求执行相应的加工中心加工程序，加工中心对半成品加工，加工完成之后，RFID 写入，更新工件状态，PLC 调度机器人去加工中心取件，机器人夹取成品进行入库，RFID 写入，更新工件状态，此时机器人处于等待状态，如果还有工件需要加工，PLC 会继续调度机器人执行下一个加工任务，直到所有工件加工完毕。

在虚拟调试的过程中，不可避免地会出现一些问题，如图 11 所示，第一个问题是机器人往数控车床上料，此时数控车床的安全门处于关闭状态，导致机器人直接撞击在安全门上；第 2 个问题是机器人往加工中心放件，加工中心的卡盘处于紧闭状态，导致工件无法正常放入；第 3 个问题是机器人直接绕过了数控车床，将毛坯工件送到了加工中心；第 4 个问题是数控车床加工完成后，机器人没有去取件。

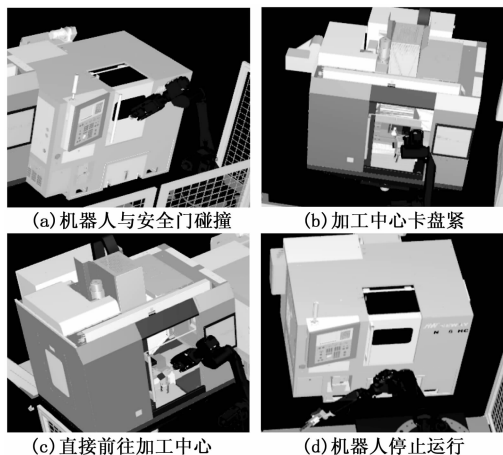


图 11 工作站虚拟调试问题

上述问题如果发生在工作现场，会损坏加工设备及或者造成加工过程无序。经过检查，前两个问题出现的原因是 PLC 梯形图程序设计存在问题，机器人往数控车床上料之前没有给安全门开信号，往加工中心上料之前没有给卡盘松信号。后两个问题出现的原因是 RoboDK 中的信号初始化及地址发送错误，数控车床的加工完成信号初始化为已完成状态，导致机器人提前往加工中心上料；数控车床加工完成之后，发出的加工完成信号的地址不是 PLC 梯形图程序中设定的加工完成信号地址。针对问题的具体原因，进行了 PLC 梯形图程序的改正及 RoboDK 中正确的初始化及信号的匹配，最后虚拟调试中，工作站能够按照工序正

常的运行，工作站正常运行如图 12 所示。

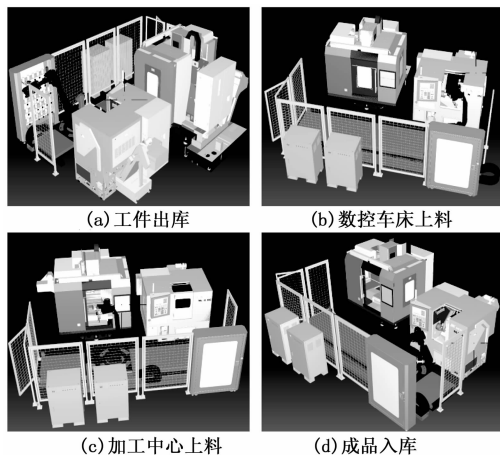


图 12 工作站虚拟调试过程图

4 实验验证

通过虚拟调试对系统进行了稳定性和可靠性测试，在工厂搭建了智能制造加工系统的实验平台进行实验。实验平台包括 PC 机、控制柜、立体仓库、汇博机器人、机器人夹爪、RFID 读写器、KND 数控车床及加工中心。其中 PC 机负责下载机器人和 PLC 程序，控制柜中包括 SIMATIC S7-1200 和汇博机器人控制器，RFID 读写器负责对立体仓库对应仓位中芯片进行读写，更新工件状态。系统实验平台如图 13 所示。

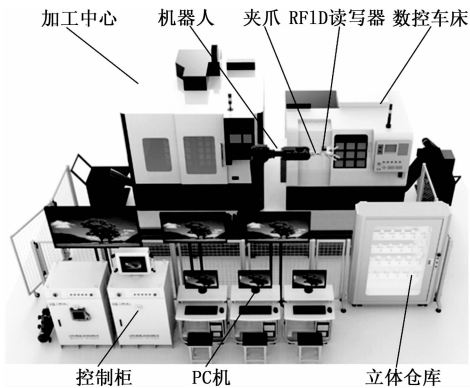


图 13 系统实验平台

将 PC 机、PLC、机器人、RFID 读写器、数控车床及加工中心接入同一交换机下的局域网中，并分配相应的 IP 地址，在机器人示教盒中载入机器人程序，操作模式调成自动并且启动机器人，PC 机通过博途软件载入 PLC 梯形图程序，启动智能制造加工系统，运行流程和虚拟调试中一样，系统运行实验如图 14 所示。其中数控车床和加工中心信号在虚拟调试中是通过工业以太网与其它设备交互的，省去了与 PLC 的硬件 IO 接线，在实验场景中，主控 PLC 配置了 2 个 16 入/16 出的继电器输出型的输入/输出模块，分别用于与数控车床及加工中心进行信号交互。系统运行中

(下转第 212 页)