

基于 RabbitMQ 与 PostgreSQL 的微组装生产线数据采集系统

史荣昌 张邱松 周洪坤

南京电子技术研究所 江苏南京 210039

DOI: 10.32629/ems.v8i7.21196

[摘要] 针对微组装生产线设备异构协议多、产线数据隔离与高性能写入等需求,使用 LabVIEW 程序开发工具,设计并实现一套多协议采集、RabbitMQ 消息队列解耦和 PostgreSQL 按产线 Schema 隔离存储的数据采集系统。系统收集设备实时状态、物料批次、工艺参数、设备利用率四类信息,采用按生产线独立数据库模式的存储方案提升并发性能。经实际产线长期运行验证,系统可稳定支撑多产线并发采集,存储和查询效率均能满足生产追溯与实时监控要求。

[关键词] 微组装;数据采集;RabbitMQ;PostgreSQL;LabVIEW

一、引言

微组装生产线是半导体器件封装和集成的重要环节,涉及贴片、键合、封盖等多类自动化设备。这些设备通常来自不同厂商,通信协议各异,包括 SECS/GEM、日志文件型、TCP 自定义协议及 PLC 等。为支撑生产过程追溯、设备综合效率分析和工艺优化,需要实时、准确采集设备状态、物料批次信息、工艺参数。原有采集方案针对单一协议或单产线设计,难以适应多产线扩展要求;而直接写入关系数据库的紧耦合架构在大流量下极易产生写入瓶颈和锁竞争。

本文提出一种基于 LabVIEW 开发采集客户端、RabbitMQ 构建消息中枢、PostgreSQL 实现存储的分层解耦系统。系统设计统一消息封装规范和四类业务队列,支持日志、SECS/GEM、TCP 和 PLC 四类采集器的无缝接入;数据表设计兼顾关系与非关系数据,从数据利用角度优化字段,提高数据可用性;采用单数据库多 Schema 的隔离模式,通过优化数据库结构,实现高吞吐、低成本的水平扩展。

二、系统总体架构

系统整体分为采集层、消息队列层、处理存储层和数据服务层四层。采集层由分布部署在产线服务器、工控上位机上的 LabVIEW 程序,每种通讯协议封装为独立的采集器程序,负责将原始数据按统一格式封装并发送至 RabbitMQ。消息队列层根据业务主题建立四类队列,通过同一交换机实现消息路由。处理存储层由一组 LabVIEW 开发的后台消费者服务程序组成,完成消息解析和数据库写入。数据服务层实现通过上层 MOM(制造运营系统)接口,自动上传物料信息、工艺参数,并提供信息名称映射关系编辑功能,根据设备状态信息,实现各产线设备利用率数据看板。

LabVIEW 作为工业自动化领域的开发平台,提供丰富的驱动和通信组件,适合直接与设备交互。RabbitMQ 具备高吞吐、灵活路由和可靠确认机制,作为数据总线,降低了采集端程序的网路维护复杂度。PostgreSQL 凭借开源、Schema

隔离、JSONB 等特性,在兼顾写入性能、非结构化字段存储查询方面优于传统时序库,且运维成本低。所有采集端与 RabbitMQ 之间采用异步发布确认模式,确保信息不丢失,消费者端使用手动 ACK 与死信队列,保证数据处理安全性。

三、多源异构设备采集模块

3.1 日志文件采集

部分设备仅通过文本日志输出信息,且一般日志文本是根据日期时间或加工产品来建立文件。对此类设备的采集,可将设备端的日志所在文件夹设置为 FTP 目录,采集程序在服务器端监控目录文件的变化,读取新的日志,抽取时间、工艺参数、状态、物料信息、产线信息等,解析后按不同的队列写入 RabbitMQ。

在确定不影响设备运行的前提下,读取过的日志文件,可按照解析成功或失败的结果分类,直接转移到不同的文件夹下。这样处理,使日志文件夹只留有未读取的文本文件,即便于后续分析采集中的问题,又使采集程序容易识别增量文件。

3.2 TCP 服务采集

针对支持 TCP 连接,传输自定义文本协议的设备,采集程序实现了基于定时轮询结构的 TCP 客户端,按预设时间发送询问指令,对设备返回信息解析处理后,通过不同的队列写入 RabbitMQ。

例如:采集程序按预设通讯端口建立 TCP 连接后,发出报文“EquipmentStatus?”,设备上位机返回报文:

```
{
  "method": 3
  "code": "STOP"
  "data": "设备故障"
}
```

解析后产生设备实时状态信息,附加当前时间信息,写入 RabbitMQ 的设备实时状态队列,完成一次状态采集。每 3~

5秒询问一次, 根据业务时间调整询问时间间隔。

系统将此通讯类型设备使用单一程序合并采集, 询问报文、返回报文解析规则、通讯端口、IP地址等差异信息作为程序参数本地存储, 实现动态添加、显示和管理。

3.3 SECS/GEM 采集

半导体设备普遍遵循 SECS/GEM 标准。本系统采用自研 LabVIEW 的 SECS/GEM 底层通讯模块, 建立采集程序与设备的协议连接。在建立连接后, 采集程序向设备发出自定义的关键参数报告 (S2F33), 定义事件与报告的触发关系 (S2F35), 启用相关事件 (S2F37)。设备在运行过程中, 每发生一次相应的事件, 会自动将报告发送到采集程序, 采集程序解析报告后写入 RabbitMQ。此种采集方式采用事件驱动, 较轮询模式效率高。

3.4 PLC 采集

对于无上位机, 只有 PLC 控制器的设备, 系统通过工业网关作为采集设备, 将各种型号的 PLC 设备信息转换, 统一使用 Modbus-TCP 通讯协议进行上传服务器。在具备无线条件的车间, 可使用带有无线功能的工业网关实现数据上传。工业网关的使用, 可以作为 IP 地址转换器, 解决设备 PLC 的 IP 网段不同、IP 地址冲突的问题; 可以将原始点位信息, 转换为统一名称、格式的规则化字段, 降低了采集程序的复杂度, 便于采集端集中统一处理。

四、基于RabbitMQ的消息队列体系设计

RabbitMQ 是一款基于 AMQP (高级消息队列协议) 的开源消息中间件, 具有高可靠、高并发、可集群扩展的特点。其核心概念包括:

- 交换器 (Exchange): 接收采集程序发送的消息, 并根据路由规则将消息转发到对应的队列。
- 队列 (Queue): 消息的存储和投递载体, 可设置持久化确保重启不丢失。

- 绑定 (Binding): 交换器与队列之间的路由规则, 支持 direct、fanout、topic 等模式

本系统选用 RabbitMQ 作为数据总线, 主要基于以下考虑

- 采集与处理解耦: 采集器无需关心后端处理能力, 消息缓冲可削峰填谷, 避免数据丢失。
- 灵活的路由能力: Topic 交换机支持按产线、消息类型等维度多级匹配, 轻松实现多业务队列分流。
- 完善的可靠性机制: 发布确认、消息持久性、手动应答和死信队列等组合, 可保障数据在软硬件故障下不丢失
- 多语言客户端: LabVIEW 可通过 .NET 接口调用 RabbitMQ 官方 .NET 客户端, Python、C# 消费者也方便接入。

4.1 核心队列功能

根据业务需求, 建立以下持久队列:

设备实时状态队列: 接收设备当前状态信息, 如运行、待机、故障等, 消费者将其更新到数据库最新状态表, 供实

时监控使用。

物料批次信息队列: 接收装配物料批次号信息, 包括载具 ID、操作员、器件位号等, 主要供生产追溯使用。

工艺参数信息队列: 接收温度、压力、真空等工艺配方参数以及重量、尺寸等检验参数信息, 供质量追溯使用。

状态流水信息队列: 专门用于接收设备状态持续时长。消息不是直接的状态值, 而是设备在变更到新状态事件发生时, 上一状态的开始、结束时刻与持续时长。

4.2 消息格式与路由

消息内容采用 JSON 定义, 示例如下:

```
{
  "assetNumber": "320114894132"
  "state": "Stop"
  "warningCode": "1165"
  "warningInfo": "气压低"
  "createTime": "20250403 09:08:57"
}
```

使用 Topic 交换机, 绑定键格式为“消息类型. 产线名称”。上述示例的绑定键为“equRealtimeStatus.SMT_line01”, 即 SMT01 生产线的设备实时队列。

对于工艺参数, 不同设备有不同的信息, 有温度曲线、贴片压力、喷印距离等等, 消息中使用 JSON 单层嵌套格式, 示例如下:

```
{
  processParameters:
  {
    "温度曲线": "200; 250; 300; 350"
    "贴片压力": 267
    "喷印距离": 20
  }
}
```

除嵌套内工艺参数外, 所有 JSON 格式消息的键都与数据库中的表名一致, 省去存储处理程序的映射关系维护。

4.3 可靠性机制

信息发布使用 publisher confirm, 消费者手动 ACK, 在存储处理程序确认存储动作完成后, 消息才会删除。系统的 RabbitMQ 为所有队列配置了死信交换机和死信队列, 处理无法解析或处理失败的消息。

五、转存储服务与数据库设计

5.1 消费者架构

处理存储层由 LabVIEW 开发的程序实现, 按队列划分独立进程, 在进程内每条产线的设备对应一个独立的消费者线程, 从队列消费消息后按顺序处理。LabVIEW 从队列中取消息, 以事件回调方式处理, 当队列有消息进入时, 处理程序收到通知可以立即处理, 避免轮询操作, 运行效率高。

5.2 核心队列存储处理

设备实时状态队列: 接收设备当前状态信息, 根据设备编号, 使用 UPDATE 更新 equRealtimeStatus 表, 维持最新的状态。状态信息时效性高, 队列消息按即时消息处理, 不做延时处理。

物料批次信息队列: 接收装配物料批次号信息, 使用 INSERT 插入 materialInfo 表, 一行记录对应一个物料的信息。为降低频繁操作数据库, 处理程序等待累计满 50 条记录或等待满 5 分钟后, 做一次批量插入数据操作; 物料批次号是上层 MOM 系统所需重要信息, 在 MOM 中, 物料批次信息需要与产品编号关联。因为微组装产品拼版生产的特殊性, 设备生产输出的信息没有产品编号, 只有载具编号, 处理程序需要从 MOM 系统中, 根据载具编号与产品编号的映射关系, 获取产品编号、订单号, 才能产生完整可用的物料批次记录。

工艺参数信息队列: 接收工艺配方参数及检验参数信息, 使用 INSERT 插入 processParameters 表。与物料批次相同, 采用批处理记录模式, 并需要建立工艺参数与产品编号的关联关系。processParameters 键, 使用 PostgreSQL 数据库的 JSONB 型字段进行存储, 实现关系型与非关系型数据的合并处理。

状态流水信息队列: 专门用于接收设备状态持续时长, 使用 INSERT 插入 equStatusFlowLog 表。消息不是直接的设备状态值, 而是设备在变更到新状态事件发生时, 上一状态的开始、结束时刻与持续时长的记录。

5.3 产线模式 (Schema) 隔离

为每个产线在 PostgreSQL 数据库中创建独立模式 (Schema), 公共数据 (设备台账、系统配置) 保留在 public 模式下。系统采用此方案, 具备以下优势: 一是数据物理隔离, 单个产线查询、备份和清理数据不影响其他产线; 二是在高并发写入时, 减少跨产线索引和锁竞争; 三是当增加新产线时, 只需复制库结构即可创建新的 Schema 及对应表, 无需修改代码, 扩展成本低。

5.4 核心表结构

每个产线 Schema 下包含以下核心表:

设备实时状态表 equRealtimeStatus: 自增 ID(主键)、设备编号、设备状态、警报信息、警报代码, 创建日期时间。

物料信息表 materialInfo: 自增 ID(主键)、设备编号、订单号、产品编号、器件位号、操作员、工装号、工装顺序、物料批次、创建日期时间, 是否已上传 MOM 等。

工艺参数表 processParameters: 自增 ID(主键)、设备编号、订单号、产品编号、器件位号、操作员、工装号、工装顺序、工艺参数 JSONB、创建日期时间、是否已上传 MOM 等。

设备状态流水表 equStatusFlowLog: 自增 ID(主键)、设备编号、设备状态、状态开始时间、状态结束时间、持续

时长、创建日期时间。

六、数据服务层功能实现

数据服务层作为本系统与上层应用交互的窗口, 主要负责数据上传、信息映射及可视化展示。基于系统总体架构设计, 本层具体实现以下三大核心功能。

6.1 MOM 系统数据交互与状态标记

物料批次与工艺参数是上层 MOM 所需的重要信息。考虑到 MOM 系统不同产品对数据字段需求的差异性, 系统采取全量上传策略, 即将采集到的物料及工艺参数全部上传至 MOM 系统接口。系统根据 MOM 接口的返回结果, 在本地数据库中对相应记录进行状态标注 (如“已上传”、“上传失败”、“无需上传”等), 确保数据追溯的完整性与准确性。

6.2 异构信息名称映射配置

由于工艺参数在 MOM 系统与数据采集系统中的命名存在不一致, 系统提供了信息名称映射关系编辑功能。用户可通过可视化界面配置采集系统字段与 MOM 系统字段的映射关系, 系统在数据上传前自动完成字段转换, 解决了异构系统间的数据语义对齐问题, 降低了系统集成复杂度。

6.3 车间可视化视图开发

为应对车间生产可视化需求, 数据服务层基于四类信息, 开发了多种视图, 应对展示数据需求。通过查询数据库中的设备状态流水与实时状态表, 系统能够生成各产线设备利用率数据看板, 直观展示设备运行、待机、故障等状态分布, 为生产调度与效率优化提供数据支撑。

七、结束语

本文介绍了使用 LabVIEW 设计实现的一套基于 RabbitMQ、PostgreSQL 数据库的微组装生产线多源采集系统, 通过多采集器、四类核心业务队列、事件驱动的消息消费和按产线 Schema 隔离存储, 有效解决了设备异构协议适配、高频写入问题。将相关数据按统一模式存储, 为生产线技术研发与制造追溯提供了数据分析的可靠数据源。实际产线长期运行表明, 系统在数据吞吐、完整性、安全性和可扩展性上均能满足复杂微组装产线要求, 未来可进一步丰富采集器类型, 并探索数据湖一体式冷热分层存储, 以进一步支撑更复杂的实时分析需求。

[参考文献]

[1] 刘胜新, 赵玉洁, 曹亚琪, 胡长明 复杂电子装备多层级、多专业协同的企业智能制造体系研究与实践[J]. 智能制造, 2022, 4: 20-24

[2] 葛警军, 王更, 邢金芳 基于数据中心的能耗测试有效查询优化技术研究[J]. 科学技术创新, 2023, 24: 84-87

[3] 戴施伟, 周凌珉, 郑一泓. 基于 SSM 框架和 RabbitMQ 技术的 OJ 系统的设计与实现[J]. 计算机时代, 2022, 10: 81-85.