ServoSystemManager V1.0 总体规划

驱动组 王彬 2016.5.22

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 时间 | 作者 | 说明 |
| V1.0 | 2016.5.22 | 王彬 | 第一版本draft |
|  |  |  |  |

# 1：目的

## 1.1：为什么要做这个系统？

|  |  |
| --- | --- |
| 原因1： | 单轴驱控一体在调试方面的一些失败教训和痛苦的维护经历。 |
| 原因2： | 为驱控一体之后的快速功能更新和大幅提高对客户需求的响应速度做基础。 |
| 原因3： | 为驱控一体产品组的各位同事大幅提高工作效率，节省人力成本（例如功率板和接口板的测试） |
| 原因4： | 管理驱控一体化相关的各个产品的固件升级和管理，并能作为工厂出货测试的工具。简化工厂工人的操作难度，节省其人力成本。 |

## 1.2：这个系统要做成什么样子？

|  |  |
| --- | --- |
| 1： | 随主控制芯片自动化升级数据流部分，界面部分稍作修改。 |
| 2： | 命令行模式，支持脚本控制。 |
| 3： | 数据分析功能（例如fft等） |
| 4： | 系统仿真功能（高级功能） |
| 5： | 动态示波器功能（包括触发，测量等功能） |
| 6： | 程序固件的下载和管理 |
| 7： | 电机的参数管理，用户可以自动生成电机参数，参数写入与读取 |
| 8： | 支持各种通信接口的扩展（分层，只需更换底层驱动） |
| 9： | 界面易用，美观，大方，用户体验好（采用qt） |
| 10： | 参数自动调整功能（一键调整环路参数） |
| 11： | 模型的辨识功能（最小二乘法） |
| 12： | 高级调试功能（任意变量及参数的读写以及FPGA寄存器的操作） |
| 13： | 运动规划的实现 |
| 14： | 简单信号的产生（方波，正弦波等） |
| 15： | 实现matlab某些分析功能 |

# 2：主要工作

## 2.1：数据流的自动化

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 底层程序头文件的扫描（细节很多需要注意） |
| 2 | Xml文件的产生 |
| 3 | 与IAR联合工作获取底层程序中的任意地址（批处理） |
|  |  |
|  |  |

## 2.2：通信接口

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 抽象层的接口 |
| 2 | 底层具体驱动接口 |
|  |  |
|  |  |
|  |  |

## 2.3：指令接口

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 针对底层实现的指令进行封装给上层使用 |
| 2 | 指令的多线程保护 |
|  |  |
|  |  |
|  |  |

## 2.4：命令行模式

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 定义指令的关键字和规则 |
| 2 | 实现指令的解析器（上层），将命令行的输入转化为对应的指令发送下去 |
| 3 | 要和之前底层对应的指令兼容与共用（底层） |
|  |  |
|  |  |

## 2.5：脚本模式

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 脚本中的每一行命令应该都是命令行支持的。只是顺序的执行 |
| 2 | 要实现时间延迟函数。方便暂停操作 |
| 3 | 脚本应该和命令行共用一个解析器 |
|  |  |
|  |  |

## 2.6：数据分析功能

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 对单一信号的FFT |
| 2 | 对输入输出进行频域分析（对系统） |
| 3 |  |
|  |  |
|  |  |

## 2.7：驱控系统仿真功能

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 驱控模型的建立 |
| 2 | 仿真操作系统的建立 |
| 3 | 能完美对接实际系统的调试软件（例如能用示波器功能观测仿真曲线） |
|  |  |
|  |  |

## 2.8：动态示波器功能

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 示波器能接收实际的数据和仿真的数据（可能需要调整显示速度） |
| 2 | 示波器能够动态显示，也能够静态显示来自文本或者其它途径的数据 |
| 3 | 示波器需要具有触发（触发方向，触发前预留点数）和测量光标功能 |
| 4 | 示波器数据的保存和再次显示 |
| 5 | 示波器数据的缩放功能（数据范围太大的时候使用） |
| 6 | 用户曲线选择和高级功能曲线选择功能 |
| 7 |  |

## 2.9：程序固件的下载和管理

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 固件的下载（上层和底层的配合）需要控制中断开关 |
| 2 | 底层的flash驱动 |
| 3 | 固件的读取对比验证 |
| 4 | 固件分为基础固件和实际使用的固件两部分，基础固件管理通信和下载等部分，上电后通过基础固件看是否烧写成功（在铁电中存储对应标志），烧写成功再加载实际固件 |
| 5 |  |

## 2.10：电机参数的管理

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 驱控一体参数的自动生成，客户填入电机参数可以自动生成能运行的参数 |
| 2 | 铁电存储器的驱动 |
| 3 | 上位机配合操作铁电存储器 |
| 4 | 铁电参数的文件的下载，从铁电读取数据到文件进行比较 |
| 5 |  |

## 2.11：电机安装功能

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 客户输入电机参数，根据该参数生成能运行的参数 |
| 2 | 自动做ADC校正和找相位的过程并保存结果 |
| 3 | 客户可启动简单Jog进行运动或者开环运行（开环电压要有斜坡）。 |
| 4 |  |
| 5 |  |

## 2.12：界面操作

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 界面主框架搭建 |
| 2 | 每个独立功能界面作为库实现，可以在主框架中添加删除 |
| 3 | 每个独立功能界面对应一个数据流的库供其调用完成该界面的功能或者某些数据流的库会给多个界面进行调用。 |
| 4 | 界面的美化 |
| 5 | 整个程序的安装包生成（使用专门工具，需要配置） |

## 2.14：环路参数自整定

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 做到一键自动整定各个环路的参数（根据输入的电机参数来选择最优的参数） |
| 2 | 自整定方法研究 |
| 3 |  |
| 4 |  |
| 5 |  |

## 2.15：电机参数自整定

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 做到一键自动整定电机的参数（针对某些没有资料的电机） |
| 2 | 自整定方法研究，怎样做到基本不运动就可以整定 |
| 3 |  |
| 4 |  |
| 5 |  |

## 2.16：模型的辨识

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 通过加激励给仿真模型和实际对象来逐步完善仿真模型，迭代的过程 |
| 2 | 模型辨识方法研究。 |
| 3 |  |
| 4 |  |
| 5 |  |

## 2.17：高级调试功能

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 任意变量及参数的读写 |
| 2 | FPGA寄存器的操作 |
| 3 | 上层对应树状结构的建立 |
| 4 |  |
| 5 |  |

## 2.18：实现简单运动规划

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 在PC上做规划，通过高速的网络进行数据交互 |
| 2 | 实现运控之前的某些运动方式（看能做到多细的规划） |
| 3 | 运动规划的仿真功能 |
| 4 |  |
| 5 |  |

## 2.19：简单信号发生器功能

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 实现电流，速度等指令的方波，正弦波等发生模块 |
| 2 | 可对信号做斜坡滤波等简单处理 |
| 3 | 多种滤波器的实现仿真比较 |
| 4 | 噪声的产生 |
| 5 |  |

## 2.20：某些matlab功能的实现

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 参考matlab的某些有用功能进行实现 |
| 2 | 不依赖于matlab的安装。 |
| 3 | 数学方法和数值计算的研究 |
| 4 |  |
| 5 |  |

## 2.21：简化测试和发货

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 通过脚本的方式来实现测试一条龙服务 |
| 2 | 顺序控制，按照顺序来执行需要测试的项（全民写脚本） |
| 3 | 发货测试也是一样的。（全自动化测试）可完全集成测试程序到该软件中。 |
| 4 |  |
| 5 |  |

## 2.22：历史记录功能

|  |  |
| --- | --- |
| 序号 | 主要工作描述 |
| 1 | 日志记录故障和报警等信息 |
| 2 | 是否有必要记录操作步骤 |
| 3 |  |
| 4 |  |
| 5 |  |

# 3：主框架

## 3.1：软件层次图



## 3.2：软件工程结构

采用VS2013工程和qt工程共用的结构。Qt主要用来做界面，VS2013可以嵌入qt插件。使用vs强大的编辑功能使得开发更高效



确定程序结构的好处是多人协同编程的时候可以更好的管理分类。每个人写的代码放到对应分类的地方不会影响其它部分功能。每个功能库要对应一个测试程序来检验库是否能满足要求。