# FES 2.0系统设计文档

| 日期 | 修改记录 | 修改人 |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

目录

[FES 2.0系统设计文档 1](#_Toc438565304)

[1．概述 3](#_Toc438565305)

[2．系统功能 3](#_Toc438565306)

[二、总体设计 3](#_Toc438565307)

[1．概述 3](#_Toc438565308)

[1.1测点索引 3](#_Toc438565309)

[1.2 测点逻辑类型 3](#_Toc438565310)

[1.3 测点数据类型 4](#_Toc438565311)

[2．设计思想 4](#_Toc438565312)

[2.1 模块划分 4](#_Toc438565313)

[2.2 引擎内核模块 6](#_Toc438565314)

[2.3 引擎接口模块 8](#_Toc438565315)

[2.4 IO驱动 9](#_Toc438565316)

[三、接口说明 9](#_Toc438565317)

[3.1 接口基本规范 9](#_Toc438565318)

[3.2 内核接口 10](#_Toc438565319)

[3.2.1 内存库创建 10](#_Toc438565320)

[3.2.2 系统变量设值 10](#_Toc438565321)

[3.3驱动接口 10](#_Toc438565322)

[3.3.1 内存库关联 10](#_Toc438565323)

[3.3.2 数据模型 10](#_Toc438565324)

[3.3.3 IO设值 11](#_Toc438565325)

[3.3.4 状态接口 11](#_Toc438565326)

[3.4应用接口 11](#_Toc438565327)

[3.5消息接口 12](#_Toc438565328)

[3.5.1 消息定义 12](#_Toc438565329)

[3.5.2 消息接口 14](#_Toc438565330)

[3.5.3 调用的报警接口 14](#_Toc438565331)

[四、数据结构说明 15](#_Toc438565332)

[4.1．基本数据 15](#_Toc438565333)

[4.1.1 DIN 15](#_Toc438565334)

[4.1.2 AIN 15](#_Toc438565335)

[4.1.3 DOUT 16](#_Toc438565336)

[4.1.4 AOUT 16](#_Toc438565337)

[4.2系统变量表 17](#_Toc438565338)

[4.2.1前置系统变量表 17](#_Toc438565339)

[4.3 模型关系表 17](#_Toc438565340)

[4.3.1通道表 17](#_Toc438565341)

[4.3.2装置表 18](#_Toc438565342)

一、简介

## 1．概述

## 2．系统功能

2.1 内存库创建以及内存库读写接口

2.2 驱动加载和卸载

2.3 全数据上送SCADA

2.4 SCADA与装置之间控制交互

2.5 调试服务端

# 二、总体设计

## 1．概述

系统的测点，也可称之为变量，根据实际外部接线方式，可分为实际测点和虚拟测点，也可以称为定位变量或者非定位变量。非定位变量也称之为内存变量，内存变量分为系统变量与用户变量。

虚拟变量（属于非定位变量）不属于IO驱动。

不同数据类型（Boolean、Int、Float）的变量的OccNo是连续，虚拟变量也可以关联报警，支持画面调用显示、写历史数据等操作。

系统变量与用户变量分为两张独立的表。系统变量表中的变量的OccNo及TagName是由系统预定义的，用户不可修改的。具体系统变量可见附表。

### 1.1测点索引

所有的测点支持TagName和OccNo索引。

### 1.2 测点逻辑类型

TYPE\_AIN

TYPE\_DIN

TYPE\_AOUT

TYPE\_DOUT

TYPE\_USER

TYPE\_SYS

### 1.3 测点数据类型

|  |  |  |
| --- | --- | --- |
| **类型名称** | **说明** | **备注** |
| **Boolean** | 位 | 开关量（遥信、遥控） |
| **Char** | 有符号字节（8位） |  |
| **Byte** | 无符号字节（8位） |  |
| **Short** | 有符号字（16位） |  |
| **Word** | 无符号字（16位） |  |
| **Long** | 有符号双字（32位） |  |
| **DWord** | 无符号双字（32位） |  |
| **Float** | 单精度浮点（32位） |  |
| **Double** | 双精度浮点（64位） | 模拟量（遥测、遥调、遥脉） |
| **String** | 字符串（0结束符） | 用于用户变量(针对，系统变量中的字符串型变量，单独建立字符串池存放字符串变量的值和相应的索引，在系统变量中，只存字符串变量的索引) |
| **Array** | 定长数组 |  |

## 2．设计思想

### 2.1 模块划分

FES部分主要包括以下部分：

|  |  |  |
| --- | --- | --- |
| 序号 | 名称 | 描述 |
| 1 | 引擎内核 |  |
| 2 | 引擎接口 |  |
| 3 | IO驱动 | 负责与装置（保护测控、PLC、仪表、通信通信机等）通信 |
| 4 | 辅助调试工具 | 调试记录、报文显示及捕捉等等 |

任务模块及任务ID划分

|  |  |  |  |
| --- | --- | --- | --- |
| 任务模块名称 | ID |  | 说明 |
| 内核引擎任务 | 1 | KERNEL\_TSK |  |
|  |  |  |  |
|  |  |  |  |
| 主从同步任务 |  | DUAL\_TSK |  |
| 调试任务 |  | DBG\_TSK |  |
| 脚本计算任务 |  | EXP\_TSK |  |
| 网络邻居同步任务 |  | FRIEND\_TSK |  |
| 总线代理模块 |  |  |  |
|  |  |  |  |
| 通道1 | 10001 |  |  |
| 通道2 | 10002 |  |  |
| 通道n | 10000+n |  |  |

主从同步和邻居同步经过前置总线代理模块，发往总线

每个任务单独维护自己邮箱，定时从邮箱接收信息

模块结构图



### 2.2 引擎内核模块

引擎内核功能如下：

|  |  |  |
| --- | --- | --- |
| 序号 | 功能 |  |
| 1 | 内存库加载及创建 |  |
| 2 | IO驱动管理 |  |
| 3 | 数据主从同步 |  |
| 4 | 调试服务端 |  |
| 5 | 控制与调节 |  |
| 6 | 脚本引擎 |  |
|  |  |  |

#### 2.2.1内存库创建模块

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 说明 | 描述 |
| 1 | 系统节点表 |  |  |
| 2 | 通道表 |  |  |
| 3 | 装置表 |  |  |
| 4 | 模拟量表 |  |  |
| 5 | 开关量表 |  |  |
| 6 | 模出量表 |  |  |
| 7 | 开出量表 |  |  |
| 8 | 用户变量表 | 系统内部维护，通过脚本开放引用 |  |
| 9 | 系统变量表 | 系统自动实时更新系统变量 | 每个节点的系统变量表是**不一样** |
| 10 | 线性转换关系表 |  |  |
| 11 | 非线性转换关系表 |  |  |
| 12 | 计算表 |  |  |
| 13 | 告警表 |  |  |

1.部分变量若配置初始值，或者自动记录（自动保存），在变量初始化时应当自动赋值。

2.内存变量（系统变量及内存变量）默认品质均为好。

**2.2.1.1内存变量管理**

1. 各表中的OccNo从1开始，每个表的OccNo独立,0表示非法OccNO。
2. TagName在一个前置节点内全局唯一，模入量、模出量、开入量、开出量、用户变量、系统变量、通道、装置采用TagName描述。

**2.2.1.2 内存表关系**

通道：一个或多个使用相同通信规约的装置形成一个通道。

装置：每个通道下辖多个装置

驱动：每个通道对应一个驱动

**2.2.1.3 基于内存库的关系映射**

点映射：通道号，装置号，点号到OccNo的映射

名映射：TagName到OccNo的映射

#### 2.2.2 IO驱动管理

**2.2.2.1、驱动加载**

前置扫描通道表，根据规约类型和驱动运行方式，加载驱动，并将通道号作为参数传递至驱动

**2.2.2.2 驱动监测**

驱动运行状态监测，驱动将运行状态刷新至前置通道表

**2.2.2.3、驱动卸载**

1. 前置退出时，设置所有通道内核运行状态为退出
2. 驱动循环检测通道内核运行状态，如果为退出状态，则退出驱动

或：前置发送退出命令至驱动，驱动处理命令退出

#### 2.2.3 数据传送

更详细内容见 通信协议

1、定时扫描内存库全数据，上送至SCADA（批量传送测值）

2、上送方式：本地共享内存或SOCKET

3、主从之间的数据同步，同步的内容见下表：

|  |  |  |
| --- | --- | --- |
|  | 内容 | 备注 |
| DIN |  |  |
| AIN |  |  |
| DQ |  |  |
| AQ |  |  |
|  |  |  |

注：主从同步时，系统变量的同步方式**待定**。

#### 2.2.4调试服务端模块

1、根据调试信息分类，建立一个或多个socket服务端

2、调试信息经socket发送至调试客户端

#### 2.2.5控制与调节

实现SCADA、装置以及驱动之间的控制命令交互

1. 实现方式：控制命令交互基于共享内存队列实现；
2. 队列标识：队列以通道ID做为唯一标识，前置的命令队列的ID为0
3. 队列需求：每个通道维护一个命令队列，前置写，驱动读；前置维护一个命令队列，驱动写，前置读；SCADA写，前置读

控制、调节、设值存在以下几种情况：

1． 本节点上的模块（脚本、顺控、面板、调试）修改本地数据；

2． 本节点上的模块（脚本、顺控、面板、调试）修改远程（其他节点）数据；

3． 远程节点修改本地数据；

SetSomeValue(Fi,PointOccNo,value)

如果是本机Fi,则直接下发，否则走总线转

#### 2.2.6脚本与逻辑

1、内核支持脚本

2、脚本中关于数据的读写操作是基于前置开放接口实现

#### 2.2.7 主从冗余同步

### 2.3 引擎接口模块

引擎接口模块主要是供IO驱动类程序模块、基于FES的高级应用模块（比如：脚本、逻辑图、顺控图等等）调用。

驱动接口：驱动对内存库读写数据接口，包括获取配置数据接口。

内核接口：内核对内存库完全操作接口。

应用接口：应用对内存库读写数据接口。

命令接口：驱动，内核，应用命令交互接口。

### 2.4 IO驱动

IO驱动主要完成与智能设备的通信，驱动程序运行模式有：进程内（dll）及进程外（exe）两种模式。驱动实现为动态库形式，如果需要进程外驱动，内核将启动**代理**进程，加载该驱动，作为进程外驱动的宿主进程。

IO驱动通过解析规约文本完成智能设备的数据采集，同时完成引擎内核转发的下行控制命令。通常来说，不同的协议文本以不同的驱动实现。IO驱动程序与引擎内核通过引擎接口模块进行数据的交互，具体框架图可见下图：

驱动入口函数：KRESULT drivermain(INT32U nChannelOccNo)

驱动使用内核接口,详见下文接口说明

关于驱动的几点说明：

1. 如果本机状态为从机状态，驱动程序不进行数据的读写操作（采集AI\DI，遥控设值等等）。
2. 如果通信异常，则该装置或者通道下相关测点品质全部为BAD。
3. 如果DO\AO的品质为BAD，则不能输出测值。
4. 通信驱动检测到通道异常，必须由驱动发出自诊断的报警。
5. SOE的报警由通信驱动程序内部解析并发起。

# 三、接口说明

## 3.1 接口基本规范

1. 统一返回值KRESULT，实际返回值以指针参数形式返回
2. 所有接口基于OCCNO，提供TagName到OCCNO的转换接口

## 3.2 内核接口

### 3.2.1 内存库创建

1、内存库创建

KRESULT CreateIO();

2、内存库删除

KRESULT DestroyIO();

### 3.2.2 系统变量设值

KRESULT SetSYSValue(INT32U occNO,VARIANT vValue,INT32U nQuality);

## 3.3驱动接口

### 3.3.1 内存库关联

1、内存库打开,并通过消息向前置注册该通道

KRESULT OpenIO(INT32U nChannelOccNo);

2、内存库关闭，并通过消息向前置注销该通道

KRESULT CloseIO(INT32U nChannelOccNo);

### 3.3.2 数据模型

1.获取通道下所有的装置号

KRESULT GetDevicesInChannel(INT32U nChanelOccNO，vecotr<INT32U>& vecDevices);

2.获取装置下所有的AIN OCCNO

KRESULT GetAinsInDevice(INT32U nDeviceOccNO，vecotr<INT32U>& vecAins);

3.获取装置下所有的DIN OCCNO

KRESULT GetDinsInDevice(INT32U nDeviceOccNO，vecotr<INT32U>& vecDins);

4.获取装置地址等参数

KRESULT GetDevice(INT32U nDevOccNO,CDevice\*\* ppDevice);

5.获取通道参数

KRESULT GetChannel(INT32U nChannelOccNO,CChannel\*\* ppChannel);

6.获取AIN

KRESULT GetAIN(INT32U occNO，AIN\*\* ppAIN);

7.获取DIN

KRESULT GetDIN(INT32U occNO，DIN\*\* ppDIN);

8.获取系统变量

KRESULT GetSYS(INT32U occNO，SYS\*\* ppSYS);

### 3.3.3 IO设值

1.AIN设值

KRESULT SetAINValue(INT32U occNO,FP32 fValue,INT32U nQuality);

2.DIN设值

KRESULT SetDINValue(INT32U occNO,INT8U nValue,INT32U nQuality);

3.DOUT设值

KRESULT SetDOUTValue(INT32U nOccNo, INT8U Value, INT8U Quality);

4.AOUT设值

KRESULT SetAOUTValue(INT32U nOccNo,FP32 Value,INT8U Quality);

### 3.3.4 状态接口

1 驱动是否退出

KRESULT IsDriverQuit(INT32U nChannelOccNo);

2 前置退出

KRESULT IsFesQuit();

## 3.4应用接口

应用类接口主要为顺控、脚本等基于前置的**本地**高级应用模块服务。

1.TagName转为OCCNO

KRESULT GetOccNoByTagName(INT8S\* strTagName,INT32U\* pOccNo);

KRESULT GetOccNoByTagName(INT32U nDataType,INT8S\* strTagName,INT32U\* pOccNo);

2.AIN取值

KRESULT AppGetAinValue(INT32U nOccNo,FP64\* pValue);

3.DIN取值

KRESULT AppGetDinValue(INT32U nOccNo,INT8U\* pValue);

4.系统变量取值

KRESULT AppGetSysValue(INT32U nOccNo,VARIANT\* pValue);

## 3.5消息接口

前置引擎与IO驱动程序之间采用邮箱通信，通道号作为邮箱的标识符。可按邮箱ID作为接收地址进行邮箱通信。

消息有以下类型：

|  |  |  |
| --- | --- | --- |
| 序号 | 简介 |  |
| 1 | 调试类消息 | 强制设值、修改定值、驱动退出等 |
| 2 | IO类消息 | 遥控、遥调发给内核 |
| 3 | 告警类消息 | 驱动告警发往内核 |
|  |  |  |
|  |  |  |
|  |  |  |

### 3.5.1 消息定义

|  |  |  |  |
| --- | --- | --- | --- |
| 属性序号 | 名称 | 说明 | 备注 |
| 1 | Type | 命令类型 |  |
| 2 | Length | 命令长度 |  |
| 3 | ChannelOccNo | 目标通道OCCNO | 发往前置则该域为0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 命令序号 | 名称 | 说明 | 备注 |
| 1 | CT\_CTRL | 控制命令 |  |
| 2 | CT\_QUIT | 退出驱动 |  |
| 3 | CT\_MSG | 自定义消息 | 告警，日志等 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

报警类消息：

模拟量越限、复限、梯度限等

|  |  |  |
| --- | --- | --- |
|  | 内容 | 备注 |
| 1 | 前置节点、测点排行 |  |
| 2 | 时间 |  |
| 3 | 报警投退状态 | 是否抑制了某类告警 |
| 4 | 报警类型 | 高限、高高限… |
| 5 | 测值 | 报警发生时的测值 |
|  |  |  |
|  |  |  |

开关量变位、SOE告警

|  |  |  |
| --- | --- | --- |
|  | 内容 | 备注 |
| 1 | 前置节点、测点排行 |  |
| 2 | 时间 |  |
| 3 | 报警投退状态 | 是否抑制了某类告警（1->0 0->1） |
| 4 |  |  |
| 5 | 测值 | 报警发生时的测值 |
|  |  |  |
|  |  |  |

注：SOE与开关量变位的区别，就是SOE告警由驱动自主解析并产生告警事件上送，

普通开关量报警的告警事件则由内核完成。

（事故、故障、状变）

自诊断报警

比如：XX通道与XX装置通信异常

XX通道与XX装置通信恢复正常

XX通道与XX装置通信切换到XX网络

|  |  |  |
| --- | --- | --- |
|  | 内容 | 备注 |
| 1 | 前置节点、通道、装置排行 |  |
| 2 |  |  |
| 3 | 时间 |  |
| 4 | 报警内容（文本） |  |
| 5 |  |  |
|  |  |  |
|  |  |  |

保护的自诊断报警

操作告警（操作成功、失败）

管理类消息：

驱动管理？

用户登录（FES没有，SCADA有此类告警）

控制类：

遥控设值，

强制设值，

修改测点属性

继电保护类的遥控（选择、遥控）

节点的主从状态切换

设时（给装置设定一个时间）

|  |  |  |
| --- | --- | --- |
|  | 内容 | 备注 |
| 1 | 前置节点、测点排行 |  |
| 2 | 测点类型 | NODE\CHANNEL\DEVICE\POINT |
| 3 | 测点属性 | 测值、投退、是否抑制报警 |
| 4 | 源 | 设值的源 |
| 5 |  |  |
| 6 | 测值 |  |
|  |  |  |

### 3.5.2 消息接口

1前置下发命令至驱动

KRESULT SendCommand(INT32U nChannelOccNo,INT8u\* pCommand,INT32U nLength);

2驱动接收命令

KRESULT RecvCommand(INT32U nChannelOccNo,INT8u\* pCommand,INT32U nLength);

3驱动上送消息至前置

KRESULT SendCommand(INT8u\* pCommand,INT32U nLength);

4前置接收命令

KRESULT RecvCommand(INT8u\* pCommand,INT32U nLength);

### 3.5.3 调用的报警接口

5 自诊断

FESAPI void IoDiagAlarm(INT32U nChannleNo, INT32U nDeviceNo, INT8S tmpstr[]);

由驱动调用，驱动程序发现装置通信故障，或者通道异常时，通过调用该接口通知系统。

6 操作报警

FESAPI void IoOperAlarm(INT32U nChannleNo, INT32U nDeviceNo, INT8S tmpstr[]);

7 保护的自诊断

FESAPI void IoRelayDiagAlarm(INT32U nChannleNo, INT32U nDeviceNo, INT8S tmpstr[], TIMEPAK \* pTm);

由驱动调用，装置内部自诊断发现异常时，通过驱动程序调用该接口通知系统。

8 通用的告警

FESAPI void IoAlarmMsg(INT32U nChannleNo, INT32U nLogType, INT8S tmpstr[], TIMEPAK \* pTm);

# 四、数据结构说明

## 4.1．基本数据

### 4.1.1 DIN

|  |  |  |
| --- | --- | --- |
| 成员 | 说明 | 备注 |
| OccNo | 索引关键字，值从1开始。内部私有变量，外部模块**禁止修改**。 |  |
| BlockNo | 板内点号，辅助用索引号 |  |
| State | 状态位 |  |
| IsDefined | 是否定义，用于备用点识别 |  |
| ScanEnable | 是否投入扫描 |  |
| IsInited | 是否初始化标志 |  |
| Quality | 测值品质 |  |
| ManSet | 人工设值投退标志 |  |
| RawValue | 原始值 |  |
| Value | 处理后测值 |  |
| Name | 测点名称 |  |
| DinFunc | 内部私有回调函数，用于内部通讯及逻辑更新（CPULinker）用，外部模块**禁止修改**。 |  |

### 4.1.2 AIN

|  |  |  |
| --- | --- | --- |
| 成员 | 说明 | 备注 |
| OccNo | 索引关键字，值从1开始。内部私有变量，外部模块**禁止修改**。 |  |
| BlockNo | 板内点号，辅助用索引号 |  |
| State | 状态位 |  |
| IsDefined | 是否定义，用于备用点识别 |  |
| ScanEnable | 是否投入扫描 |  |
| IsInited | 是否初始化标志 |  |
| Quality | 测值品质 |  |
| ManSet | 人工设值投退标志 |  |
| RawValue | 原始值 |  |
| Value | 处理后测值 |  |
| Name | 测点名称 |  |
| AinFunc | 内部私有回调函数，用于内部通讯及逻辑更新（CPULinker）用，外部模块**禁止修改**。 |  |

### 4.1.3 DOUT

|  |  |  |
| --- | --- | --- |
| 成员 | 说明 | 备注 |
| OccNo | 索引关键字，值从1开始。内部私有变量，外部模块**禁止修改**。 |  |
| BlockNo | 板内点号，辅助用索引号 |  |
| State | 状态位 |  |
| IsDefined | 是否定义，用于备用点识别 |  |
| ScanEnable | 是否投入扫描 |  |
| IsInited | 是否初始化标志 |  |
| Quality | 测值品质 |  |
| IsSBO | 测点输出类型是否采用SBO方式 |  |
| RelayState | 外部继电器状态 |  |
| Output | 开出输出值映象 |  |
| Name | 测点名称 |  |
|  |  |  |

### 4.1.4 AOUT

|  |  |  |
| --- | --- | --- |
| 成员 | 说明 | 备注 |
| OccNo | 索引关键字，值从1开始。内部私有变量，外部模块**禁止修改**。 |  |
| BlockNo | 板内点号，辅助用索引号 |  |
| State | 状态位 |  |
| IsDefined | 是否定义，用于备用点识别 |  |
| ScanEnable | 是否投入扫描 |  |
| IsInited | 是否初始化标志 |  |
| Quality | 测值品质 |  |
| RawValue | 原始值 |  |
| Output | 模拟量输出值映象 |  |
| Name | 测点名称 |  |
|  |  |  |
|  |  |  |

## 4.2系统变量表

### 4.2.1前置系统变量表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| OccNo | Name | 描述 | 数据类型 | 备注 |
| 1 | $ |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |
| 5 | $Year | 年 |  |  |
| 6 | $Month | 月 |  |  |
| 7 | $Day | 日 |  |  |
| 8 |  | 时 |  |  |
| 9 |  | 分 |  |  |
| 10 |  | 秒 |  |  |
| 11 |  | 毫秒 |  |  |
| 12 |  |  |  |  |
| 13 |  |  |  |  |
| 14 |  |  |  |  |
| 15 | $MyNodeState | 当前节点工作状态 |  |  |
| 16 | $ChannelNumber | 通道数量 | **DWord** |  |

## 4.3 模型关系表

### 4.3.1通道表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 属性名称 |  |  |  |
| 1 | 通道OCCNO |  |  |  |
| 2 | 通道名称(TagName) |  |  |  |
| 3 | 通道规约名称(modbus) |  |  |  |
| 4 | 驱动运行方式(进程内，进程外) |  |  |  |
| 5 | COM |  |  |  |
| 6 | BAUD |  |  |  |
| 7 | 驱动内核状态 | 1：加载；0：卸载 | 内核设置：值为1则加载驱动，为0则卸载驱动 |  |
| 8 | 驱动运行状态 | 1：运行；0：停止 | 驱动设置：启动时设为1，退出时设为0 |  |
|  |  |  |  |  |
|  |  |  |  |  |

### 4.3.2装置表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 属性名称 |  |  |  |
| 1 | 装置OCCNO |  |  |  |
| 2 | **所属通道OCCNO** |  |  |  |
| 3 | 装置名称 |  |  |  |
| 4 | 装置地址 |  |  |  |
| 5 | IP |  |  |  |
| 6 | PORT |  |  |  |
|  |  |  |  |  |

五、SCADA与前置内存数据库设计

5.1 前置数据按数据类型分块

5.2 SCADA为每个前置维护一份数据，每份数据独立存在

5.3 点不连续，由驱动建立点地址到OCCNO的映射关系

5.4 系统变量

每个节点均有系统变量

系统变量基于共享内存按独立模块实现，在模块内部定时上送至SCADA

前置启动时一次性创建所有所需的共享内存