**成都远望科技有限责任公司**

受控

**文件编号：项目代号.XXX**

WRSOC

详细设计说明

V1.1

第1次修订

|  |  |  |  |
| --- | --- | --- | --- |
| **编制部门** | **拟制/日期** | **审核/日期** | **批准/日期** |
| **硬件产品开发部** |  |  |  |
| **会签** |  | | |
| **接收部门** | **总经理□ 副总经理□ 行政部□ 财务部□ 集成测试部□**  **质量管理部■ 远望科技产品开发部■ 远望信息产品开发部■** | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **修改日期** | **版本** | **修订次数** | **修订人** | **修订内容** |
| 1 |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**目录**

[1. 范围 1](#_Toc533684449)

[1.1. 标识 1](#_Toc533684450)

[1.2. 系统概述 1](#_Toc533684451)

[1.3. 文档概述 1](#_Toc533684452)

[1.4. 引用文件 1](#_Toc533684453)

[2. 系统结构 1](#_Toc533684454)

[3. 系统1设计 1](#_Toc533684455)

[3.1. 功能设计 1](#_Toc533684456)

[3.1.1. 功能1 1](#_Toc533684457)

[3.1.2. 功能2 3](#_Toc533684458)

[3.2. 接口设计 3](#_Toc533684459)

[3.2.1. 接口1 3](#_Toc533684460)

[3.2.2. 接口2 3](#_Toc533684461)

[3.3. 性能设计 3](#_Toc533684462)

[3.4. 运行设计 3](#_Toc533684463)

[3.5. 数据库设计 3](#_Toc533684464)

[3.6. 安全设计 3](#_Toc533684465)

[4. 系统2设计 3](#_Toc533684466)

[4.1. 功能设计 3](#_Toc533684467)

[4.2. 接口设计 3](#_Toc533684468)

[4.3. 性能设计 3](#_Toc533684469)

[4.4. 运行设计 3](#_Toc533684470)

[4.5. 数据库设计 3](#_Toc533684471)

[4.6. 安全设计 3](#_Toc533684472)

[5. 系统3设计 4](#_Toc533684473)

[5.1. 功能设计 4](#_Toc533684474)

[5.2. 接口设计 4](#_Toc533684475)

[5.3. 性能设计 4](#_Toc533684476)

[5.4. 运行设计 4](#_Toc533684477)

[5.5. 数据库设计 4](#_Toc533684478)

[5.6. 安全设计 4](#_Toc533684479)

# 范围

## 标识

## 系统概述

WRSOC系统主要由系统主机、环境监控主机、温湿度传感器、空调控制器、条形码扫描设备、短信猫设备、以及相关软件系统组成。系统通过RS485或RS232总线采集环境数据、采集UPS工作状态数据、采集精密空调工作状态数据，同时通过相关接口控制空调运行，达到调节环境温湿度的目的；通过网络监控雷达以及RDA计算机运行状态，采集雷达性能参数数据；通过网络采集视频图像数据，以及云台控制。所有采集到的数据存储到WRSOC系统主机中。

## 文档概述

本文档为WRSOC系统C++程序的详细设计说明，介绍每个模块的逻辑流程和主要接口。

## 引用文件

# 系统结构



# WRSOCcontrolPlug设计

## 功能设计

WRSOCcontrolPlug为插件的入口类，该类定义在WRSOCcontrtolMain.cpp中，该类除了负责插件的初始化，启动和销毁外，该类还负责一下几个功能。

* 创建初始化其他模块的类，并开启多个线程，在这些线程中执行各个模块的功能。
* 设置抓图定时器，定时从视频主机抓取图片，并将图片信息保存在数据库中。
* 设置磁盘监测定时器，定时监测WRSOC主机磁盘余量，若余量小于阈值则报警。
* 设置雷达文件缺失报警定时器，若在规定时间内没有生产雷达文件，则报警。
* 设置连接查询定时器，定时查询WRSOC主机与周围设备的连接状态，若连接有异常则报警。
* 设置备份定时器，定时备份文件，并清除过时文件。

WRSOCcontrolPlug中还包括一些辅助函数，主要是时间戳获取函数以及将配置项写入XML或者数据库的函数。



## 接口设计

|  |  |
| --- | --- |
| 接口 | 功能 |
| PingRadar | 通过Ping命令查询雷达连接状态。 |
| onInitialTimeout | 初始化定时器的定时函数，用于开启各个模块功能 |
| onCaptureJPEGTimeout | 抓图定时器定时函数，用于抓取图片 |
| onHDCfgTimeout | 磁盘监测定时函数，用于监测磁盘状态 |
| onRadarFileTimeout | 备份定时函数，定时备份文件，清除过时文件 |
| onRadarFileDetectTimeout | 雷达文件监测函数，定时查询雷达是否生成文件 |
| onConnectionTimeout | 连接定时函数，定时查询各设备连接状态 |
| GetFileModificationTime | 获取文件修改时间戳 |
| writeRegisterInfoToDataBase | 将注册信息写入数据库中 |
| writeH5streamToDataBase | 将H5Stream路径信息写入数据库中 |
| DVRLogin | 登录视频服务器 |
| DVRLogout | 退出视频服务器 |
| getAirConditionerStr | 从数据库中获取空调配置信息 |
| parseAirConditionStr | 解析空调配置信息字符串 |

## 性能设计

## 运行设计



## 数据库设计

该类主要涉及的数据库表格是systeminfo, radarrunstatus, thresholdparameter等。

## 安全设计

# 离线文件模块设计

## 功能设计

离线文件模块主要负责检测四种离线文件的生成，并解析文件中的内容并放入数据库中。这四类文件包括动态范围文件，相位噪音文件，发射率文件和太阳法文件。



## 接口设计

|  |  |
| --- | --- |
| 接口 | 功能 |
| loadTXTFile | 检测新的离线标定文件 |
| parseFile | 读取离线标定文件内容并解析 |
| checkFileTime | 检测离线标定文件的修改时间 |
| writeToDataBase | 将解析内容存放如数据库中 |
| setUpdateTime | 保存新的离线标定文件的最近修改时间 |

## 性能设计

## 运行设计



## 数据库设计

该模块涉及的数据库包括dyntestinfo，dyntestresult， phasenoiseinfo， phasenoiseresult， refcalibresult， sunresult。

## 安全设计

# 环境采集模块设计

## 功能设计

该模块功能主要包括环境温湿度的采集和UPS状态信息的采集。该模块主要包括CTemperature， UPSHandlerRS232，UPSHandlerSNMP四个类组成。其中CTemperature负责向环境主机发命令并获取温湿度。UPSHandlerRS232负责通过串口协议获取UPS状态信息。UPSHandlerSNMP负责通过SNMP协议获取UPS状态信息。

## 接口设计

CTemperature类主要接口如下：

|  |  |
| --- | --- |
| 接口 | 功能 |
| setID | 设置环境主机ID |
| setIPAddress | 设置发送的IP地址 |
| processCmd | 处理环境主机返回的命令 |
| startTimer | 开启定时器，定时到达后发送命令 |
| stopTimer | 关闭定时器 |
| writeToDataBase | 将温湿度写入数据库中 |

UPSHandlerRS232类主要接口如下：

|  |  |
| --- | --- |
| 接口 | 功能 |
| exec | 线程执行函数，在函数里定时发送命令 |
| processData | 解析返回的UPS状态数据 |
| startReceive | 开启网络端口 |
| stopReceive | 关闭网络端口 |
| queryUPSData | 发送UPS命令 |
| writeToDataBase | 将UPS状态信息写入数据库中 |

UPSHandlerSNMP类主要接口如下：

|  |  |
| --- | --- |
| 接口 | 功能 |
| getUPSInfo | 从数据库里读取UPS的MIB信息 |
| exec | 线程执行函数，在函数里定时发送命令 |
| queryUPSData | 发送UPS命令 |
| writeToDataBase | 将UPS状态信息写入数据库中 |

## 性能设计

## 运行设计

环境采集模块通过开启定时器或者线程，每隔一段时间采集一次数据，并保存到数据库中。

## 数据库设计

该模块涉及的数据库包括upsinfo，thresholdparameter，radarrunenvirement。

## 安全设计

# SMS短信模块

## 功能设计

该模块主要是负责将报警信息通过短信猫发送到用户的手机上。该模块包括两个类，

SMSHandler和SMSThread，SMSHandler用于保存用户的手机号和原始的短信信息，以及通过串口发送命令和接收回复。SMSThread主要负责在线程中将原始的短信信息进行处理，转化成短信编码格式后通过串口发送出去，并对异常做出响应。



## 接口设计

SMSHandler主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| StartCom | 打开串口 |
| StopCom | 关闭串口 |
| getSMSRawData | 获取短信原始信息 |
| addSMSData | 添加短信信息 |
| sendUartData | 发送串口数据 |
| sendSMSText | 在SMSThread中开启线程，准备发送短信 |

SMSThread主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| Entry | 线程入口函数，在线程里进行短信编码转换和发送 |
| processSMSData | 将原始短信信息转换为短信编码 |
| transformTelNum | 将手机号码进行转换 |
| transformContent | 将短信内容进行转换 |

## 性能设计

## 运行设计



## 数据库设计

## 安全设计

# 雷达参数文件解析模块

## 功能设计

该模块主要负责监控各个雷达型号的关键参数文件，并将其中的关键参数和适配参数读取出来并保存到数据库中。并计算出雷达健康指数。该模块包括以下CRadarParser，负责解析雷达参数中的关键参数和静态参数；CSARadarParser，CSCRadarParser，CCCRadarParser，CCDRadarParser负责解析雷达参数中的适配参数；CHealthIndex负责计算雷达健康指数

## 接口设计

CRadarParser类的主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| exec | 线程入口函数，在函数内部检测新的文件并解析 |
| getLatestFileName | 获取最新的雷达参数文件 |
| loadFile | 读取最新文件的文件内容 |
| writeSiteToDataBase | 将站址信息写入数据库 |
| writeRadarDataToBase | 将关键参数写入数据库 |
| startRadarFileDetection | 启动雷达文件缺失监测 |

CSARadarParser等类的主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| exec | 线程入口函数，在函数内部检测新的文件并解析 |
| getLatestFileName | 获取最新的雷达参数文件 |
| loadFile | 读取最新文件的文件内容 |
| writeRadarDataToBase | 将关键参数写入数据库 |

CHealthIndex类的主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| calHealthIndex | 计算健康指数 |
| calEmitterTubeFault | 判断发射管是否异常 |
| calEmitter3A5Fault | 判断3A5是否异常 |
| calPinzhongFault | 判断中频是否异常 |
| calReceiverADFrontFault | 判断AD前端是否异常 |
| calReceiveADEndFault | 判断AD后端是否异常 |
| calAntennaBeamFault | 判断天线束是否异常 |

## 性能设计

## 运行设计



## 数据库设计

雷达参数解析模块涉及的数据库包括calibrationparameter，onlinemonitorpoint，adaptparameter，radarconstparameter。

## 安全设计

# FTP上传模块

## 功能设计

该模块只有一个类FTPThread，用于将雷达参数文件、离线标定文件等文件上传到指定服务器。并将上传失败的文件信息写入数据库中

## 接口设计

FTP上传模块主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| Entry | 线程入口函数，在该函数中上传文件 |

## 性能设计

## 运行设计



## 数据库设计

## 安全设计

# 网络服务模块

## 功能设计

网络模块主要负责与设备之间进行网络通信的功能，主要包括与Java后台的通信、与雷达的通信、与空调的通信等，具体功能如下：

* 接收Java后端发送的雷达控制命令，然后转发给雷达；定时通过Ping来监测雷达连接是否正常，若正常则发送雷达状态查询命令，并解析返回的状态信息然后存入数据库。
* 接收Java后端发送的空调控制命令，包括普通空调和精密空调，接收后解析命令，然后想空调发送相应的控制命令。
* 接收Java后端发送的短信信息，短信信息包括手机联系人号码和短信内容，接收到短信信息后交给短信模块进行处理和发送。

该模块包括CNetService，负责与Java后端通信，所有Java后端发送的命令都通过这个类接收；CRadarController，该类负责向雷达发送控制命令，并定时接收雷达返回的状态信息；CAirConditionerController和CAirHandler负责解析普通空调命令并向普通空调发送控制命令；CSpecAirConditionController和CSNMPAirConditionController负责向精密空调发送命令。



## 接口设计

CNetService类的主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| startReceive | 打开网络端口 |
| stopReceive | 关闭网络端口 |
| sendDataToFront | 将数据发送给java后端 |
| sendDataToAirConditioner | 发送命令到普通空调 |
| sendDataToRadar | 发送命令到雷达 |
| setAirConditionerCmdFlag | 设置空调是否接收到数据标识位 |
| processData | 处理从java后端发送的命令 |
| reStartReceive | 端口号更改后重新启动网络 |

CRadarController类的主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| startReceive | 打开网络端口 |
| stopReceive | 关闭网络端口 |
| sendDataToRadar | 将命令发送给雷达 |
| cmdCompare | 判断当前雷达状态信息是否与上次相同 |
| setStatusStr | 保存当前雷达状态信息 |
| parseStatusStr | 解析雷达状态信息 |
| writeToDataBase | 将雷达信息保存在数据库中 |
| reStartReceive | 端口号更改后重新启动网络 |

CRadarController和CAirHandler类的主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| processData | 接收空调返回的命令信息 |
| stopReceive | 关闭网络端口 |
| startReceive | 打开网络端口 |
| sendDataToAirConditioner | 发送命令到普通空调 |
| setairConditionFlag | 设置空调收到命令后的标识位 |
| parseAirCmd | 解析从java端收到的空调控制命令 |
| fabricateSwitchOnCmd | 根据空调协议组建开机命令 |
| fabricateSwitchOffCmd | 根据空调协议组建关机命令 |
| fabricateModeCmd | 根据空调协议组建模式控制命令 |
| fabricateTemperatureCmd | 根据空调协议组建温度设置命令 |

CSpecAirConditionController和CSNMPAirConditionController类的主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| processData | 接收精密空调返回的状态信息 |
| stopReceive | 关闭网络端口 |
| startReceive | 打开网络端口 |
| startTimer | 开启定时器定时发送精明空调状态查询命令 |
| setairConditionFlag | 设置空调收到命令后的标识位 |
| parseAirCmd | 解析从java端收到的空调控制命令 |
| parseAirStatus | 解析从空调返回的状态信息 |
| onAirStatusTimeout | 定时器函数，定时发送指令 |
| sendSwitchCmd | 发送开关机命令 |
| sendTemperatureCmd | 发送温度控制命令 |
| sendHumidityCmd | 发送湿度控制命令 |

## 性能设计

## 运行设计







## 数据库设计

网络服务模块涉及的数据库包括radarrunstatus，radarcontrolinfo，airconditioninginfo。

## 安全设计

# Alarm报警模块

## 功能设计

报警模块主要负责对读取到的雷达关键参数进行阈值判断，若超出阈值则报警，并将报警信息写入数据库中。另外还监控雷达的报警文件日志，发现有新的报警文件后读取文件并解析报警信息，并将报警信息写入数据库中。

该模块主要包括CLogAlarm，负责监控报警日志文件并解析；CRadarStatusAlarm负责对雷达参数进行阈值判断。

## 接口设计

CRadarStatusAlarm类的主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| getAlarmType | 从数据库中获取报警类别 |
| getAlarmLevel | 从数据库中获取报警等级 |
| getAlarmCNname | 从数据库中获取报警内容 |
| getThresholdValue | 从数据库中获取阈值范围 |
| RadarStatusCheck | 对雷达参数进行阈值判断 |

CLogAlarm类的主要接口如下

|  |  |
| --- | --- |
| 接口 | 功能 |
| alarmFileMonitor | 线程入口函数，在线程中监测报警文件并解析 |
| getAlarmType | 从数据库中获取报警类别 |
| getLatestFileName | 获取最新的报警文件 |
| getAlarmInfo | 将报警信息保存到成员变量中 |
| parseAlarmInfo | 解析报警文件内容 |
| writeToDataBase | 将报警信息写入数据库中 |

## 性能设计

## 运行设计





## 数据库设计

网络服务模块涉及的数据库包括thresholdparameter，alarmconstinfo，alarminfo。

## 安全设计