|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 软盘编号 |  | 密级 |  | | | |
|  |  |
| CAD |  | 阶段  标记 |  |  | S |  |
|  |  |
| 会签 | |  | | --- | |  | | **分机设备监控与设备接口软件--项目总结报告** | |  |  |  |  | | --- | --- | | 编 写 |  | | 校 对 |  | | 审 核 |  | | 标 审 |  | | 批 准 |  |   **北京瀛创数字科技有限公司** | | | | | |
|
|
|
|
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|
|
| 描图 |
|  |
| 描校 |
|  |
| 旧底图登记号 |
|  |
| 底图登记号 |
|  |
|  |

目录

[1 范围 1](#_Toc495659004)

[1.1 标识 1](#_Toc495659005)

[1.2 系统概述 1](#_Toc495659006)

[1.3 文档概述 1](#_Toc495659007)

[2 任务来源与研制依据 1](#_Toc495659008)

[3 软件概述 2](#_Toc495659009)

[4 软件研制过程 3](#_Toc495659010)

[4.1 软件研制过程概述 3](#_Toc495659011)

[4.2 软件需求分析 4](#_Toc495659012)

[4.3 软件设计 4](#_Toc495659013)

[4.4 软件集成和测试 5](#_Toc495659014)

[5 软件关键技术 5](#_Toc495659015)

[5.1 可配置网络通讯协议解析模块 5](#_Toc495659016)

[5.2 数据接口协议利用xml配置 6](#_Toc495659017)

[5.3 面向对象 6](#_Toc495659018)

[5.3.1 面向对象的基本概念 6](#_Toc495659019)

[5.3.2 面向对象开发方法的思想 7](#_Toc495659020)

[5.4 分治法 7](#_Toc495659021)

[5.5 应用git管理项目代码 7](#_Toc495659022)

[6 软件满足任务指标情况 8](#_Toc495659023)

[7 配置管理情况 8](#_Toc495659024)

[8 测试和验证情况 8](#_Toc495659025)

[9 结束语 9](#_Toc495659026)

# 范围

## 标识

* 1. 已批准的标识号：YC-CASC-0001-SDMS-020
  2. 标题：分机设备监控与设备接口软件项目总结报告
  3. 本文档的适用范围：分机设备监控与设备接口软件项目情况的总结汇报工作

## 系统概述

本项目由基带TMTC模式系统监控软件、基带TTC模式系统监控软件、其他设备监控软件、远程系统监控软件组成，硬件环境与软件环境参见以下列表：

**表1-1 环境配置表**

|  |  |
| --- | --- |
| **环境** | **配置** |
| 硬件环境 | 处理器：Intel(R) Core(TM) i7-2715QE；CPU：2.10GHz；内存：8.00GB； |
| 软件环境 | 采用中标麒麟64位操作系统环境；  集成开发环境为Qt5.5.1版本； |

## 文档概述

分机设备监控与设备接口软件的开发工作已经完成。写此项目总结报告，以方便指导我们在以后的项目开发中取得更好的进步，让我们在今后的项目开发中有更多的有据资料来规范我们的开发过程和提高我们的开发效率，从而创造更好的软件项目。

# 任务来源与研制依据

本项目经航天长征火箭技术有限公司（以下称甲方）与北京瀛创数字科技有限公司（以下称乙方）协商，甲方委托乙方进行分机设备监控与设备接口软件的部分功能进行研制开发，并对研制内容、技术要求、质量控制及验收要求等达成协议进行软件的研发工作。

本项目研制依据以下文档：

1. 《分机设备监控与设备接口软件技术协议》文档；
2. 《分机设备监控与设备接口软件--需求规格说明书》文档；

# 软件概述

监控接口设备用于地面系统数据采集、分析、任务状态模拟验证，设备包含基带、上变频器、下变频器、上行开关矩阵、下行开关矩阵等。

分机设备监控与设备接口软件，为监控接口设备提供配套软件，主要用于监控设备状态，设备包括基带TMTC监控设备软件、基带TTC模式监控设备软件、其它设备监控软件和远程监控设备软件4个模式的配置项，对其进行业务数据的处理，对设备进行控制和任务的模拟验证等，软件组成如下图所示：

监控/远程分机设备监控与设备接口软件

基带TMTC模式

基带TTC模式

其它设备监控

数传模式

单遥控模式

USB模式

扩频模式1

扩频模式2

图 1 软件组成示意图

设备监控软件通过网络接受TMTC模式模拟测试验证设备、TTC模式模拟测试验证设备发送的工作状态及数据信息，在计算机显示器以图形用户界面（GUI）方式显示，同时按照约定协议向TMTC模式模拟测试验证设备、USB模式模拟测试验证设备下发设备工作参数及数据（文件）。软件部署以及使用要求如下图所示：

远程监控计算机

分机监控计算机

分机设备

分机设备

分机设备

图 2 软件部署示意图

分机设备监控软件可以部署在分机监控计算机上，通过网络及分机监控接口来监控各个设备。远程监控软件部署在一台计算机上，通过网络与监控计算机连接，通过远程监控接口操作监控计算机上相对应的设备来达到远程监控效果。

# 软件研制过程

## 软件研制过程概述

针对本项目的软件研制过程，以时间轴线划分为软件需求分析、软件需求评审、软件系统设计、软件的代码编写与功能实现、软件的集成和测试。具体过程如下所示：

1. 2017-01-10至03-10 进行需求调研以及分析，并形成相关文档；
2. 2017-03-11至03-30 进行软件系统设计；
3. 2017-04-01至07-15进行项目开发工作，有效代码110967行，程序运行所需配置文件60余个；
4. 2017-07-15至07-24进行项目首轮内部测试与回归；
5. 2017-07-25至08-05进行联合测试与回归，累计执行测试用例700个；
6. 2017-08-05 进行压力测试；
7. 2017-08-06至08-10进行120小时不间断拷机项目验收测试。

## 软件需求分析

一月十号至三月十日对测站测试验证设备远程控制软件业务需求与功能需求进行了详细的分析，软件分为如下几个功能模块：

1. 设备状态的显示：监控计算机通过网络接收来自设备的状态信息，并通过分机接口的处理后，发送到监控软件，监控软件随后将其显示在软件的GUI界面上。
2. 设备状态参数的设置：通过修改监控软件参数数值，并将其通过分机监控接口处理成设备软件可识别的报文发送到分机设备软件，从而使设备修改其状态参数。
3. 参数配置功能：提供基带TMTC模式监控设备软件、基带TTC模式监控设备软件工作的参数装订界面，并为远程监控软件共用，支持即时参数和任务宏参数设置，用户设置工作参数后通过网络将控制命令下发至某设备软件对控制命令进行解析并执行完成参数配置。
4. 远程监控的实现：通过远程计算机发送指令将分机监控设备文件中的信息元同步到远控本地，然后远控设备通过远控接口解析出设备的状态数值显示到界面上，远控端控制设备依然如此。

## 软件设计

根据《测站测试验证设备远程监控软件技术协议》以及《测站测试验证设备远程监控软件--需求规格说明书》进行软件的技术选型与整体架构设计。

软件的开发设计我们采用Qt编程语言,Qt编程语言具有以下优点：

1. 优良的跨平台性，支持Windows、Windows NT、Linux、Solaris、SunOS、HP-UC、Unix等主流操作系统；
2. 面向对象，Qt的良好封装机制使得Qt的模块化程度非常高，可重用性较好，对于用户开发开说非常方便；
3. 丰富的API，Qt包括多达250个以上的C++类；
4. 支持2D/3D图形渲染，支持OpenGL；
5. 大量的开发文档与示例。

分机/远程设备监控软件的设计内容有如下几点:

1. 通讯方式采用UDP通信，通讯接口符合《远程监控计算机的网络接口协议V1[1].00（公开）》文档的要求规范；
2. 多线程处理数据的接收与发送；
3. 为减少数据丢包情况出现数据接收时采用缓存机制；
4. 数据接口协议利用xml进行配置；
5. 界面效果利用qt自带的qss实现；
6. 软件具有中英双版，利用qt自带的语言替换机制+配置文件实现不同语言的切换。

## 软件集成和测试

测站测试验证设备远程监控软件根据《TMTC系统监控配置项测试说明》 、《TMTC系统监控配置项测试说明》 、《TTC和TMTC系统监控配置测试说明公共部分》 、《TMTC远程监控配置项测试说明》 、《TMTC远程监控配置项测试说明》 、《TTC和TMTC远程监控配置测试说明公共部分》文档进行系统化测试，测试出来的BUG录入到Redmine项目管理系统中进行闭环管理并统计。

分机设备监控软件在对BUG有效的管理与更改下，在甲方组织的120小时不间断拷机过程中稳定运用无异常问题产生。

# 软件关键技术

## 可配置网络通讯协议解析模块

通过对本项中接触到的网络接口协议，我们进行整理与分析发现现在所使用的接口协议具有极高的可复用情况，但是在不同的项目中可能会存在某个字段会有细微的调整。这种情况的出现会使代码版本的控制付出很大代价，为减少次代价在今后的项目中发生，我司积极的组织项目组成员以及技术部人员进行讨论，经多次讨论以及技术验证最终形成了《可配置网络通讯协议解析模块》可执行方案，并在本项目中使用。

可配置网络通讯协议解析模块主要是在不改动任何代码的情况下，利用与其相关联的配置文件解析网络数据包，解析后的网络数据包中的每一个参数都会具有一个规则性的识别ID。其他业务功能都可以根据识别ID进行取用。

可配置网络通讯协议解析模块与传统的网络协议解析方式比较具有如下优势：

1. 代码维护与版本控制代价小；
2. 协议解析灵活，例如新项目中需要将上一版本接口协议中的参数int A，改为double A并且保留5位有效小数。传统的协议解析方式需要改动一处或多处代码，代码更改完成后还需要重新进行编译，而可配置网络通信协议解析模块并不需要这么麻烦只需要更改配置文件并重启程序即可;
3. 在后续的升级版本中将加入可视化配置管理工具，提高新协议的生成与维护效率。

## 数据接口协议利用xml配置

将数据接口协议通过xml配置，统一管理，降低了工作的繁琐程度，减小了后期维护的难度。

## 面向对象

### 面向对象的基本概念

对象、类、实例、继承、封装、多态等。

1. 对象指的是一个独立的、异步的、并发的实体，可定义为由一组数据信息和施加于该组数据上的一组操作行为组成的集合体。可以将对象视为一个拟人化的客观存在的实体，该实体是有其结构特性和行为特征来描述的。它能理解一些事情即存储数据；做一些工作既提供服务；与其它对象协同既交换信息。
2. 类是一组具有相同结构特征和行为特征的对象，所有这些对象都是类的实例。一个类的不同实例必定具有：相同的操作集合；相同的信息结构、属性定义；不同的对象标志。实例是某个具体的对象。类是表现对象结构共性和行为的一种定义、机制或模版，类中至少包含了两方面的描述；本类所有实例的属性定义或结构定义；本类所有实例操作的定义。
3. 继承性，若B继承类A，那么，类A中的所有属性和方法均成为类B的组成部分，同时成为类B为类的子类，类A为类B的父类。继承性的软件重用的核心思想。
4. 多态性，多态既一个名词可具有多种语句，含义有两方面：同一对象接收到不同的信息采用不同的行为方式；同一消息可以根据发送消息对象的不同采用多种不同的行为方式
5. 封装是一种信息隐藏技术，用户只能见到对象封装界面上的信息对象内部对用户是隐藏的，封装的目的是将对象的的使用者和对象的设计者分开、使用者不必知道行为实现的细节，只需用设计者提供的消息来访问对象。封装定义为：一个清楚的边界，所有对象的内部软件的范围闲的在此边界以内；一个接口，此接口表述这个对象和其它对象之间的相互作用；受保护的内部实现，实现给出了由软件对象提供的功能实现细节。

### 面向对象开发方法的思想

1. 采用面向对象的概念和方法，以贴近人类自然的思维方式分析、抽象、表达系统开发过程中各方面的信息。
2. 采用模型化的手段，将系统的开发过程视为由用户需求为起点的用户模型到测试模型为终点的一系列模型转化过程，以进一步抽象和简化系统。
3. 将系统开发过程划分为分析、构造、测试等阶段，合理的组织与管理各类开打人员与用户。
4. 面相对象的技术可以保证软件的可复用性。

## 分治法

分治法所能解决的问题一般具有以下几个特征：

1. 该问题的规模缩小到一定的程度就可以容易地解决
2. 该问题可以分解为若干个规模较小的相同问题，即该问题具有最优子结构性质。
3. 利用该问题分解出的子问题的解可以合并为该问题的解；
4. 该问题所分解出的各个子问题是相互独立的，即子问题之间不包含公共的子问题。

## 应用git管理项目代码

Git是目前国际流进的分布式版本控制工具，可以管理各种项目。在项目的开发过程中，需要将代码在不同的地点复制在不同的地方，并行调试，这相，我们应该git的对代码进行同步。Git可以保存从前的修改，所以也从前根本上杜绝了误删除导致返工的问题。Git是开源的，是麒麟系统自带的所以非常安全。

# 软件满足任务指标情况

根据《分机设备监控与设备接口软件--需求规格说明书》中的提到的性能要求，本软件均已满足，具体要求如下：

1. 可以监控计算机，接收、显示并控制设备状态。
2. 可以通过远程计算机，接收、显示并控制设备状态。
3. 软件无异常、卡死或崩溃现象发生。
4. 状态查询频率1秒/次；
5. CPU占用率在50%以下，软件界面对用户操作没有可察觉的延迟；
6. 软件启动时间小于1分钟；

# 配置管理情况

测站测试验证设备远程监控软件Git代码版本管理工具、Redmine项目管理工具以及《软件开发代码规范》保证软件在整个软件开发的生命周期中各个阶段都能得到精确的项目配置。

项目组各成员利用Git工具对本项目进行了迭代开发，在开发过程中代码同步率在99%以上，同步错误的代码也可利用工具进行及时更正。

# 测试和验证情况

测站测试验证设备远程监控软件根据项目管理中BUG跟踪管理规范以及Redmine项目管理工具中的BUG管理功能进行科学有效的控制与管理，并最终的每一个BUG形成闭环。

本项目共设计条测试用例700条测试用例，测试之初在我方与甲方组织的联合测试组定义了一个较为完善的测试流程，该流程如下：

1. 实施用例；
2. 记录测试结果；
3. 描述BUG现象以及BUG产生原因；
4. 更改BUG；
5. 回归测试；
6. 关闭BUG。

依照制定好的流程我们共同努力下完成了全部的测试用例，初次测试共产生90条BUG。根据流程对每一个BUG都达到闭环效果，使其逐步归0。

在经过多轮的测试与分析后，最终在由军方组织的系统120小时不间断拷机测试过程中稳定运用无异常问题产生。

# 结束语

经过团队历时8个月的努力，分机设备监控与设备接口软件各项指标都达标，通过了军检拷机测试，可以交付使用。