|  |  |
| --- | --- |
| **文件编号** |  |
| **保密级别** |  |

**系统设计说明书**

[1.系统整体设计 2](#_Toc523215646)

[1.1.系统架构图 2](#_Toc523215647)

[1.2.模块列表 3](#_Toc523215648)

[2. 子模块概要 3](#_Toc523215649)

[2.1. 协议转换层 3](#_Toc523215650)

[2.1.1功能模块列表 3](#_Toc523215651)

[2.1.2错误消息及错误码设计 3](#_Toc523215652)

[2.2. 连接管理 4](#_Toc523215653)

[2.2.1功能模块列表 4](#_Toc523215654)

[2.3. 签名认证 4](#_Toc523215655)

[2.3.1功能模块列表 4](#_Toc523215656)

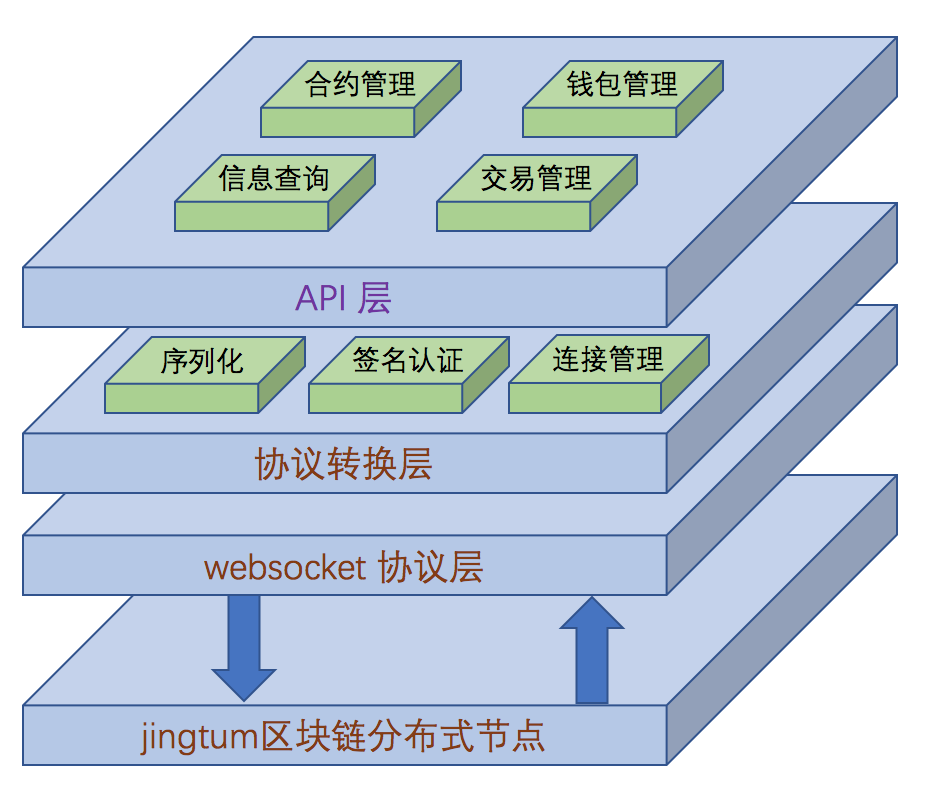
[2.4. 序列化 4](#_Toc523215657)

[2.5. API层 5](#_Toc523215658)

[2.5.1功能模块列表 5](#_Toc523215659)

# 1.系统整体设计

## 1.1.系统架构图



## 1.2.模块列表

|  |  |
| --- | --- |
| 模块名称 | 功能概要 |
| websocket协议层 | 与区块链节点进行交互的底层协议。 |
| 协议转换层 | 使用websocket发送请求、接收返回数据。  负责将返回值间返回给对应的请求。 |
| 连接管理 | 管理数据连接（创建、关闭）。 |
| 签名认证 | 提供base58、ecdsa、secp256k1、k256等常用加密码工具，并提供本地签名功能。 |
| 序列化 | 提供jingtum区块链协议要求的报文序列化功能等。 |
| API层 | 封装了对外提供的所有API，主要包括：合约管理、钱包管理、信息查询、交易管理。 |

# 子模块概要

## 2.1. 协议转换层

### 2.1.1功能模块列表

|  |  |  |
| --- | --- | --- |
| 模块名称 | 实现类 | 功能概要 |
| WebSocket封装 | WebSocket | 实现WebSocket接口，完成信息的发送与接收，基于facebook开源的SocketRocket来实现 |
| 请求管理 | HandleProcessTask | 为请求生成标识，请绑定处理器 |

### 2.1.2错误消息及错误码设计

*概要设计时确定。*

## 2.2. 连接管理

### 2.2.1功能模块列表

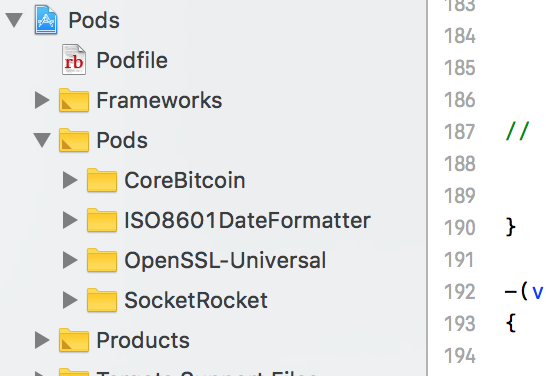
|  |  |  |
| --- | --- | --- |
| 模块名称 | 实现类 | 功能概要 |
| 连接管理 | Remote | 对外提供数据发送、关闭功能。 |

## 2.3. 签名认证

### 2.3.1功能模块列表

提供base58、ecdsa、secp256k1、k256等常用加密码工具，并提供本地签名功能，基于github上的 CoreBitcoin、OpenSSL等实现的

相关工具类如下：



**本地签名过程描述**：

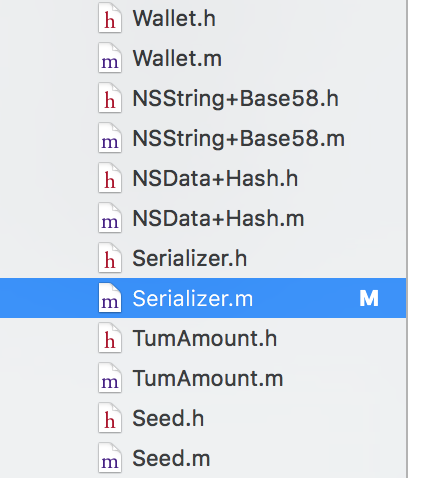
为了保护私钥安全，需要将信息在本地进行签名，然后将签名后的信息传到jingtum系统。签名的过程是：

1. 整理数据，将基础货币除以1000000，设置公钥。
2. 将数据以key/value对的形式序列化成二进制。
3. 将序列化结果取hash512。
4. 对hash进行ECDSA签名，并将签名结果放到数据中。
5. 对更新后的数据再进行序列化。
6. 将序列化的结果发送到jingtum系统。

## 2.4. 序列化

提供jingtum区块链协议要求的报文序列化功能等。

相关工具类如下：



## API层

封装了对外提供的所有API，主要包括：合约管理、钱包管理、信息查询、交易管理。

### 2.5.1功能模块列表

|  |  |
| --- | --- |
| 模块名称 | 功能概要 |
| 合约管理 | 封装合约的部署、管理功能。 |
| 钱包管理 | 封装钱包的生成、签名、校验等操作。 |
| 信息查询 | 负责链上信息的查询功能。 |
| 交易管理 | 交易相关操作。 |