**JSON-RPC二次开发接口文档V1.0**

**引言**

本档案用于对AP8000V2上位机采用jsonrpc2.0协议进行二次开发的接口说明

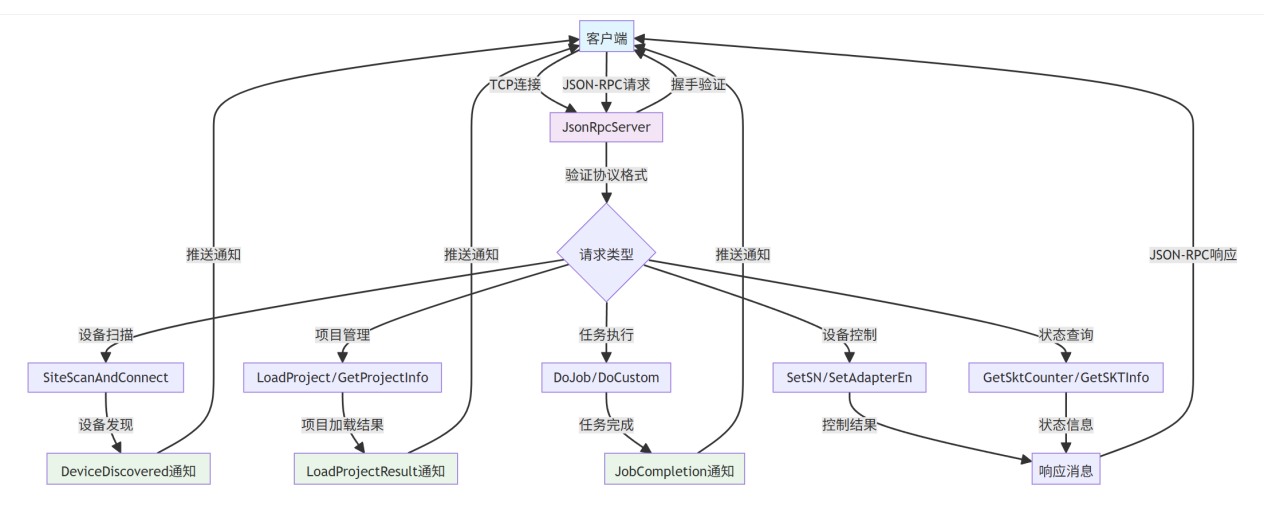
|  |  |
| --- | --- |
| 作者 | 李柯林 |
| Email | likelin@acroview.com |
| 公司 | [www.acroview](http://www.acroview).com |
| 版本 | V1.0 |
| 时间 | 20250523 |
| 修改备注 | V1.0初稿 |

后续版本升级过程时，将文件夹内容全部覆盖即可。

与软件包一起的还有ACSvcDemo.exe，这个是Acroview提供的基于qt5.15.2开发的Demo程序。后面会对该Demo程序进行介绍.

# 2 介绍

## 2.1 系统交互流程图



整体软件部署

客户通过jsonrpc接口进行自己软件的开发与AP8000V2作为服务器接收客户程序执行并执行相应的动作和返回必要的信息。

#### 大小端模式约定

大端模式，是指数据的高字节保存在内存的低地址中，而数据的低字节保存在内存的高地址中，这样的存储模式有点儿类似于把数据当作字符串顺序处理：地址由小向大增加，而数据从高位往低位放；这和我们的阅读习惯一致。

小端模式，是指数据的高字节保存在内存的高地址中，而数据的低字节保存在内存的低地址中，这种存储模式将地址的高低和数据位权有效地结合起来，高地址部分权值高，低地址部分权值低。

示例，比如0x12345678整个4个字节的整数，从地址0开始存放

如果按照大端方式存放，则地址0-3字节中的内容为0x12 0x34 0x56 0x78

如果按照小端方式存放，则地址0-3字节中的内容为0x78 0x56 0x34 0x12

前面加上0x表示该数为16进制数。

后续命令参数描述中，请注意数据的存放描述。避免因为数据的放置方式错误或者解析错误导致问题的出现。

### 2.1.1协议说明

传输协议: TCP

端口: 12345（默认需要调用Start方法指定）

连接方式: 长连接

数据格式: JSON-RPC 2.0

要与此服务器成功通信，客户端必须发送遵循以下结构的数据包：

##### 1.二进制头部(32字节)格式:

|  |  |  |  |
| --- | --- | --- | --- |
| 4字节 MagicNumber | 2字节版本号 | 4字节数据长度 | 22字节保留字段 |
| JSON-RPC 2.0 消息内容（UTF-8编码） | | | |

MagicNumber: 0x4150524F ("APRO")

版本号: 1

数据长度: JSON消息的字节长度

保留字段: 22字节，填充为0

2. JSON-RPC 2.0 消息体 (Payload)

这部分是UTF-8编码的JSON字符串，其长度必须与头部中的 "Payload Length" 字段完全匹配。JSON对象必须遵循JSON-RPC 2.0规范。

### 2.1.2JSON消息体示例

请求 (Request) - 需要服务器响应

当您需要服务器返回结果时，必须提供一个非空的 `id`。

例如，调用 "LoadProject" 方法:

{

"jsonrpc": "2.0",

"method": "LoadProject",

"params": {

"path": "C:/projects/my\_project",

"taskFileName": "task.json"

},

"id": 1

}

## 2.2 API方法详细说明

### 2.2.1 设备管理接口

开启服务，Aprog.exe -r

通用请求:{

"jsonrpc": "2.0",

"method": "", string方法

"params": {}, jsonobject参数

"id": "1" (可选)

}

通用响应:{

"jsonrpc": "2.0",

"method": "", string方法

"result": {} jsonobject参数

}

通用错误响应:

{

"jsonrpc": "2.0",

"method": "Error",

"result": { 里面包含各种异常信息

}

}

#### SiteScanAndConnect/setSiteReuslt -

#### 扫描并连接设备/获取站点信息

描述: 启动设备扫描，发现网络中的测试设备

请求:

{

"jsonrpc": "2.0",

"method": "SiteScanAndConnect",

"params": {},

"id": "1"

}

响应/setSiteReuslt:

一个site回应一条setSiteReuslt信息

{

"jsonrpc": "2.0",

"method": "setSiteReuslt",

"result":

{

"DPSFPGAVersion": "0x2002285A",

"DPSFwVersion": "V2.2.0006A",

"FPGALocation": "Normal",

"FPGAVersion": "0x2030121A",

"FirmwareVersion": "2.02.031",

"FirmwareVersionDate": "20250111",

"MUAPPVersion": "2.02.029",

"MUAPPVersionDate": "20240111",

"MULocation": "Normal",

"MacAddr": "00:3A:22:00:01:01",

"MainBoardInfo": {

"HWVersion": "0100-010000-010000-010000",

"OEM": "ACVIEW",

"SN": "A06U24120700012",

"UID": "CCE3DF00A00C0672"

},

"SiteAlias": "Site02",

"UDPPort": "8080"

}

}

### 2.2.2项目管理接口

#### LoadProject/setLoadProjectResult/setSKTEnResult -

#### 加载项目/获取项目信息结果/获取使能信息结果

命令说明：请求加载工程，该命令为异步执行，调用之后服务器启动线程执行加载工程的操作，函数立即返回，加载结果通过SetLoadResult事件通知客户端。

请求:

path:AP8000V2(MT分支)项目文件路径

taskFileName:项目文件本体

{

"jsonrpc": "2.0",

"method": "LoadProject",

"params": {

"path": "C:\\Users\\Administrator\\Desktop\\xt422\\AIPE\\Build\\task",

"taskFileName": "project.actask"

},

"id": "2"

}

setLoadProjectResult响应:

result中的字段说明:

data:{}一个jsonobject里面包含nHop,strIP两个字段

nHop:跳数 strIP：站点的ip

proInfo:用一个jsonobject包含项目信息里面含有:

doCmdSequenceArray 数组jsonArray；特别注意，下面的dojob命令中的 命令序列填充就是把数组中的其中一个jsonObject填充进去,doJob命令执行的前提一定是获取到了此数组，并且数组中有对应的操作序列

pro\_chipdata string类型项目的芯片信息，驱动信息等；

pro\_url项目的文件路径，

result:当前站点的加载结果，用于快速判定成功还是失败，布尔类型

{

"jsonrpc": "2.0",

"method": "setLoadProjectResult",

"result": {

"data": {

"nHop": 0,

"strIp": "192.168.31.30"

},

"proInfo": {

"doCmdSequenceArray": [

{

"CmdID": "1801",

"CmdRun": "Erase",

"CmdSequences": [

{

"ID": "801",

"Name": "Erase"

},

{

"ID": "803",

"Name": "BlankCheck"

}

],

"CmdSequencesGroupCnt": 2

},

{

"CmdID": "1803",

"CmdRun": "Blank",

"CmdSequences": [

{

"ID": "803",

"Name": "BlankCheck"

}

],

"CmdSequencesGroupCnt": 1

},

{

"CmdID": "1800",

"CmdRun": "Program",

"CmdSequences": [

{

"ID": "807",

"Name": "Erase If BlankCheck Failed"

},

{

"ID": "803",

"Name": "BlankCheck"

},

{

"ID": "800",

"Name": "Program"

},

{

"ID": "802",

"Name": "Verify"

}

],

"CmdSequencesGroupCnt": 4

},

{

"CmdID": "1802",

"CmdRun": "Verify",

"CmdSequences": [

{

"ID": "802",

"Name": "Verify"

}

],

"CmdSequencesGroupCnt": 1

},

{

"CmdID": "1804",

"CmdRun": "Secure",

"CmdSequences": [

{

"ID": "804",

"Name": "Secure"

}

],

"CmdSequencesGroupCnt": 1

},

{

"CmdID": "1806",

"CmdRun": "Read",

"CmdSequences": [

{

"ID": "806",

"Name": "Read"

}

],

"CmdSequencesGroupCnt": 1

},

{

"CmdID": "1901",

"CmdRun": "Self",

"CmdSequences": [

],

"CmdSequencesGroupCnt": 0

}

],

"pro\_chipdata": {

"bDebug": true,

"bottomBoard": "",

"chipAdapter": "BGA153(11.5x13)-P050-G16-01",

"chipAdapter2": "NULL",

"chipAdapter3": "NULL",

"chipAdapterID": "48A9",

"chipAlgoFile": "Drv-XSA300.drv",

"chipAppFile": "burn.app",

"chipBufferSize": 1000,

"chipBufferSizeHigh": 0,

"chipChipInfo": "bbbb.html",

"chipCurSbk": "NULL",

"chipDrvParam": 1,

"chipFPGAFile": "MUX64.jbc",

"chipFPGAFile2": "G8eMMC-TGeneric.jbc",

"chipHelpFile": "NULL",

"chipId": "0",

"chipIdACXML": "",

"chipModifyInfo": "NULL",

"chipMstkoFile": "Mst-XSA300.drv",

"chipName": "XSA300D",

"chipOperCfgJson": "{\"baseOper\":{\"blank\":true,\"blockProg\":true,\"erase\":true,\"function\":false,\"illegalBit\":false,\"prog\":true,\"read\":true,\"secure\":true,\"verify\":true},\"bitsOper\":{\"bit12\":false,\"bit16\":false,\"bit4\":false,\"bit8\":false},\"checkSumOper\":{\"crc16\":false,\"crc32\":false,\"sum\":false,\"wordSum\":false},\"fileLoadOper\":{\"bigEndian\":false,\"wordAddress\":false},\"otherOper\":{\"EEPROM\":false,\"IDCheck\":false,\"addressRelocate\":false,\"compare\":false,\"emptyBuffer\":false,\"enableSN\":false,\"insection\":false,\"loopFun\":false,\"masterCopy\":false,\"online\":false,\"pin\":false,\"protect\":false,\"unTest\":false,\"vccVerify\":true}}",

"chipOperateConfigMask": 0,

"chipPackage": "LCC16",

"chipProgType": 83886083,

"chipSbkId": 0,

"chipSectorSize": 1048576,

"chipStatus": "",

"chipType": "Other",

"manufacture": "XT",

"nVersion": 0

},

"pro\_url": "C:/Users/Administrator/Desktop/xt422/AIPE/Build/project/default.eapr"

},

"result": true

},

"id": "load\_001"

}

setSKTEnResult响应:

{

"jsonrpc": "2.0",

"method": "setSKTEnResult",

"result": {

"data": {

"data": [

],

"results": [

"BPUEn:3855 recvIP:192.168.31.30 nHop:0"

]

},

"status": 1

}

}

### 2.2.3 任务执行接口

#### DoJob/setDoJobResult - 执行标准任务

描述: 执行预定义的标准任务

参数说明：

"sktEn": 需要烧录的座子映射图

"strIP": 站点的 ip地址

"docmdSeqJson"：填入setLoadProjectResult中获取的jsonArray中的一项jsonObject

请求:

{

"jsonrpc": "2.0",

"method": "DoJob",

"params": {

"docmdSeqJson": {

"CmdID": "1800",

"CmdRun": "Program",

"CmdSequences": [

{

"ID": "807",

"Name": "Erase If BlankCheck Failed"

},

{

"ID": "803",

"Name": "BlankCheck"

},

{

"ID": "800",

"Name": "Program"

},

{

"ID": "802",

"Name": "Verify"

}

],

"CmdSequencesGroupCnt": 4

},

"sktEn": 2,

"strIP": "192.168.31.30"

},

"id": "3"

}

DoJob响应setDoJobResult:

参数说明

BPUID:BPU的索引

SKTidx:座子的索引

nHopNum: 跳数

result:执行dojob的结果

strip:执行dojob的站点信息

{

"jsonrpc": "2.0",

"method": "setDoJobResult",

"result": {

"BPUID": "B0",

"SKTIdx": 0,

"nHopNum": 0,

"result": "Passed",

"strip": "192.168.31.30"

},

"id": "3"

}

#### DoCustom/setDocustomResult - 执行自定义任务

描述: 执行用户自定义的复杂任务

参数说明:

docustomData:自定义数据，支持二进制和json

sktEn:需要执行自定义任务的座子映射图

strIP:需要执行自定义任务的站点的ip

请求:

{

"jsonrpc": "2.0",

"method": "DoCustom",

"params": {

"doCustomData": {

"cmd": "test"

},

"sktEn": 4,

"strIP": "192.168.31.30"

},

}

setDocustomResult响应:

参数说明:

bpuid:来自下位机的哪个bpu的索引

data:自定义的数据上传

ip:来自下位机的站点的ip

result:用于快速判定dojob的命令执行是否成功

{

"jsonrpc": "2.0",

"method": "setDocustomResult",

"result": {

"bpuid": 1,

"data": "abcdefghijk1234567890",

"ip": "192.168.31.30",

"result": true

},

}