

**SPRS组件**

测试文档

|  |  |  |
| --- | --- | --- |
|  | 当前版本： | V4.0.0 |
| 文件状态： | 创 建 人： | 翟海青 |
| [√]草稿 | 创建日期： | \_\_\_2020\_年 5 月 19 日 |
| [ ]正在修改 | 审 核 人： | 翟海青 |
| [ ]正式发布 | 审核日期： | \_\_\_2020\_年 5 月 23 日 |
|  | 密 级： | ■内部资料 □公开资料 |

版权声明和保密须知：

本文件出现的任何文字叙述、文档格式、照片、插图、方法、过程等内容，除特别声明外，版权均归属北京源清慧虹信息科技有限公司所有，受到有关产权及版权法保护。

Copyright© 2014.All rights reserved 北京源清慧虹信息科技有限公司

目录

[1. 说明 3](#_Toc41062738)

[2. 设备管理 5](#_Toc41062739)

[2.1. 增加节点 5](#_Toc41062742)

[2.2. 删除节点 6](#_Toc41062743)

[2.3. 修改节点 6](#_Toc41062744)

[2.4. 查询节点 7](#_Toc41062745)

[3. 子设备访问 8](#_Toc41062746)

[3.1. 查询指令表信息 8](#_Toc41062748)

[3.2. 获取设备信息 9](#_Toc41062749)

[3.3. 获取数据 10](#_Toc41062750)

[3.4. 自定义指令 11](#_Toc41062751)

[3.5. 透传指令 11](#_Toc41062752)

# 说明

总线支持的设备数量：定义为247。设备表大小：4\*N+1，N为总线支持的设备数量。

使用Test组件对SPRS组件进行测试时遵循以下规则：

在Shell交互中，固定输入“ test sprs [ ] ”，中括号内为参数，在以下会进行详细说明

SPRS组件共支持以下几种命令

enum SUB\_FUNCTION\_CODE

{

    ADD\_EQ\_NODE=1,          //增加节点

    DEL\_EQ\_NODE=2,          //删除节点

    CHA\_EQ\_NODE=3,          //修改节点

    GET\_EQ\_NODE=4,          //查询节点

    GET\_EQ\_INDEX\_INFO=9,    //获取指令表索引信息

    GET\_EQ\_CFG\_INFO=10,     //获取配置信息

    GET\_EQ\_GET\_SDAT=11,     //获取采样数据

    EQ\_CUSTOM\_COMMAND =12,  //自定义指令

    SEN\_EQ\_PAS     =13,     //透传指令

};

SPRS组件的输入接口：

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **示例（业务组件下行消息）-现行** | | | | | | | | |
| mq\_type | dat\_len | dat[0] | dat[1] | dat[2] | dat[3] | dat[4] | … | dat[n] |
| 1：增加节点 | 4 | 增加的节点地址 | 节点类型 | 节点属性 | 节点分组 | NULL | … | NULL |
| 2：删除节点 | 1 | 删除的节点地址 | NULL | NULL | NULL | NULL | … | NULL |
| 3：修改节点 | 4 | 修改的节点地址 | 节点类型 | 节点属性 | 节点分组 | NULL | … | NULL |
| 4：查询节点 | 1 | 查询的节点地址 | NULL | NULL | NULL | NULL | … | NULL |
| 9：查询指令表 | 0 | NULL | NULL | NULL | NULL | NULL | … | NULL |
| 10：获取配置信息 | 1 | 地址 | NULL | NULL | NULL | NULL | … | NULL |
| 11：获取数据 | 1 | 地址 | NULL | NULL | NULL | NULL | … | NULL |
| 12：自定义 | 1 | 地址 | NULL | NULL | NULL | NULL | … | NULL |
| 13：透传指令 | n+1 | D0 | D1 | D2 | D3 | D4 | … | Dn |

对应的数据结构体为：

//SPRS组件数据/命令输入结构

typedef struct \_\_attribute\_\_ ((\_\_packed\_\_))

{

    uint8\_t  mq\_type;                       //消息类型

    uint8\_t  dat\_len;                       //数据长度

    uint8\_t  dat[256];

}sprs\_in\_struct\_type;

以增加节点为例：

sprs\_in\_struct.mq\_type=ADD\_EQ\_NODE;   //命令

sprs\_in\_struct.dat\_len=4;             //数据长度

sprs\_in\_struct.dat[0]=addr;           //数据0

sprs\_in\_struct.dat[1]=type;           //数据1

sprs\_in\_struct.dat[2]=par;            //数据2

sprs\_in\_struct.dat[3]=gro;            //数据3

rt\_memset(&test\_tmp\_dmgms, 0, sizeof(DM\_GMS\_STRU));

mb\_make\_dmgms(&test\_tmp\_dmgms,0,&sem\_test,CP\_CMD\_DST(ADD\_EQ\_NODE),MB\_STATN\_SPRS,MB\_STATN\_TESTCOMP,(uint8\_t \*)&sprs\_in\_struct,sprs\_in\_struct.dat\_len+2,&resp\_test); //向多维消息体中装入消息

上面展示了一个封装数据的实例。

SPRS组件的输出接口：

对应的结构体为：

/\*SPRS组件返回消息结构体 \*/

typedef struct \_\_attribute\_\_((\_\_packed\_\_))

{

  uint8\_t src;           //消息源

  uint8\_t eq\_addr;       //传感器地址

  uint8\_t eq\_type;       //传感器类型

  uint8\_t eq\_par;        //传感器属性

  uint8\_t eq\_gro;        //传感器分组

  uint8\_t mq\_type;       //指令类型

  uint8\_t is\_mq\_success; //0成功，错误代码请查看错误规范

  uint8\_t d\_len;         //返回的数据长度

  uint8\_t \*dp;           //返回的数据指针

} sprs\_output\_dat\_type;

因此需要对这些命令进行测试，SPRS回复消息时，如果执行成功，测试组件会先打印ACK，然后打印RES值，如果执行失败，则只打印ACK。

测试条件：总线上挂载一台地址为1的HCF710。

测试说明：由于总线上挂在的设备种类多种多样，测试组件未对返回的数据进行解析。

# 设备管理



## 增加节点

输入：test sprs add\_node#addr#type#par#gro

示例：test sprs add\_node#3#2#0#0

说明：增加节点（地址=3，类型=2，附加属性无，分组无），地址输入范围1-247；

程序检查项：

1：设备表是否存在；

2：设备表是否能够正常打开；

3：地址是否在支持的范围内；

4：设备表中改位置的地址是否为0，不为0则不能添加；

5：设备表数量是否已经到达最大值

6：输入的设备类型参数是否正确。

返回：

添加成功：

LOG\_CMD\_ACK:{

"ut":946697097.897,"cpname":"sprs","cmd":"ADD\_EQ\_NODE",

"status":"suc","err\_code":"0x00"

}

LOG\_CMD\_RES:{

"ut":946707488.991,"cpname":"sprs","cmd":"ADD\_EQ\_NODE",

"value":0x03 0x02 0x00 0x00 }

添加失败：

LOG\_CMD\_ACK:{

"ut":946696859.211,"cpname":"sprs","cmd":"ADD\_EQ\_NODE",

"status":"err","err\_code":"0xe3"

}

## 删除节点

输入：test sprs del\_node#addr

示例：test sprs del\_node#3

说明：删除地址号为3的设备，地址输入范围0-247，当地址为0时，删除全部节点；

程序检查项：

1：设备表是否存在；

2：设备表是否能够正常打开；

3：地址是否在支持的范围内；

4：设备表中改位置的地址是否为0，为0则不能删除；

5：设备表数量是否为0，为0，这说明总线上无设备，不能进行删除操作。

删除成功：

LOG\_CMD\_ACK:{

"ut":946696990.671,"cpname":"sprs","cmd":"DEL\_EQ\_NODE",

"status":"suc","err\_code":"0x00"

}

LOG\_CMD\_RES:{

"ut":946696990.690,"cpname":"sprs","cmd":"DEL\_EQ\_NODE",

"value":0x00 0x00 0x00 0x00 }

删除失败：只返回ACK

LOG\_CMD\_ACK:{

"ut":946697063.487,"cpname":"sprs","cmd":"DEL\_EQ\_NODE",

"status":"err","err\_code":"0xe5"

}

## 修改节点

输入：test sprs cha\_node#addr#type#par#gro

示例：test sprs cha\_node#3#2#0#0

说明：修改节点（地址=3，类型=2，附加属性无，分组无），地址输入范围1-247；

程序检查项：

1：设备表是否存在；

2：设备表是否能够正常打开；

3：地址是否在支持的范围内；

4：设备表中改位置的地址是否为0，为0则不能修改；

5：输入的设备类型参数是否正确。

修改成功：

LOG\_CMD\_ACK:{

"ut":946707727.507,"cpname":"sprs","cmd":"CHA\_EQ\_NODE",

"status":"suc","err\_code":"0x00"

}

LOG\_CMD\_RES:{

"ut":946707727.518,"cpname":"sprs","cmd":"CHA\_EQ\_NODE",

"value":0x03 0x03 0x00 0x00 }

修改失败：

LOG\_CMD\_ACK:{

"ut":946707782.418,"cpname":"sprs","cmd":"CHA\_EQ\_NODE",

"status":"err","err\_code":"0xe7"

}

## 查询节点

输入：test sprs inq\_node#addr

示例：test sprs inq\_node#3

说明：查询节点（地址=3），地址输入范围1-247；

程序检查项：

1：设备表是否存在；

2：设备表是否能够正常打开；

3：地址是否在支持的范围内；

4：设备表中改位置的地址是否为0，为0返回错误。

查询成功：

LOG\_CMD\_ACK:{

"ut":946707923.27,"cpname":"sprs","cmd":"GET\_EQ\_NODE",

"status":"suc","err\_code":"0x00"

}

LOG\_CMD\_RES:{

"ut":946707923.37,"cpname":"sprs","cmd":"GET\_EQ\_NODE",

"value":0x03 0x03 0x00 0x00 }

查询失败：

LOG\_CMD\_ACK:{

"ut":946707966.411,"cpname":"sprs","cmd":"GET\_EQ\_NODE",

"status":"err","err\_code":"0xe8"

}

# 子设备访问



## 查询指令表信息

输入：test sprs get\_inx

示例：test sprs get\_inx

说明：查询指令表信息块，信息块共16字节，用于放版本、支持的设备种类等信息。

程序检查项：

1：指令表是否存在；

2：读出的指令块大小是否为16；

查询成功：

LOG\_CMD\_ACK:{

"ut":946708024.569,"cpname":"sprs","cmd":"GET\_EQ\_INDEX\_INFO",

"status":"suc","err\_code":"0x00"

}

LOG\_CMD\_RES:{

"ut":946708024.580,"cpname":"sprs","cmd":"GET\_EQ\_INDEX\_INFO",

"value":0x04 0x00 0x01 0x03 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 }

查询失败：

LOG\_CMD\_ACK:{

"ut":946707966.411,"cpname":"sprs","cmd":"GET\_EQ\_NODE",

"status":"err","err\_code":"0xC1"

}

## 获取设备信息

输入：test sprs get\_cfg#addr

示例：test sprs get\_cfg#1

说明：通常用于查询设备的保持寄存器，地址输入范围1-247；

程序检查项：

1：指令表是否存在；

2：读出的指令块大小是否为16；

3：所查询的设备地址不为0；

4：指令表打开正常；

5：读取的指令块大小正常；

6：子设备超时未响应。

7：子设备返回的数据中，返回的地址不对或CRC校验不对。

获取配置成功：

LOG\_CMD\_ACK:{

"ut":946708761.317,"cpname":"sprs","cmd":"GET\_EQ\_CFG\_INFO",

"status":"suc","err\_code":"0x00"

}

LOG\_CMD\_RES:{

"ut":946708761.328,"cpname":"sprs","cmd":"GET\_EQ\_CFG\_INFO",

"value":0x01 0x03 0x20 0x00 0x01 0x00 0x00 0x41 0x42 0xfe 0x5d 0x41 0x42 0xea 0x17 0x00 0x14 0x02 0x00 0x41 0x1c 0xd2 0xf2 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x41 0x20 0x00 0x00 0x10 0x49 }

获取配置失败：

LOG\_CMD\_ACK:{

"ut":946710564.363,"cpname":"sprs","cmd":"GET\_EQ\_CFG\_INFO",

"status":"err","err\_code":"0xd2"

}

## 获取数据

输入：test sprs get\_dat#addr

示例：test sprs get\_dat#1

说明：通常用于获取子设备的采样数据，SPRS会根据指令表中的设备指令类型（单、双指令），自动执行单/双指令数据获取，地址输入范围1-247。

程序检查项：

1：指令表是否存在；

2：读出的指令块大小是否为16；

3：所查询的设备地址不为0；

4：指令表打开正常；

5：读取的指令块大小正常；

6：子设备超时未响应。

7：子设备返回的数据中，返回的地址不对或CRC校验不对。

获取配置成功：

LOG\_CMD\_ACK:{

"ut":946710588.181,"cpname":"sprs","cmd":"GET\_EQ\_GET\_SDAT",

"status":"suc","err\_code":"0x00"

}

LOG\_CMD\_RES:{

"ut":946710588.192,"cpname":"sprs","cmd":"GET\_EQ\_GET\_SDAT",

"value":0x01 0x04 0x26 0x00 0x00 0x0e 0x6d 0x01 0x22 0x41 0xd3 0x00 0x00 0x3e 0xa7 0xcc 0x8b 0x43 0x03 0x00 0xea 0x41 0x47 0x11 0x0b 0x3e 0x84 0xde 0x80 0x41 0x46 0xee 0x4b 0x3e 0x7b 0xfb 0x80 0x41 0xa0 0x76 0xd2 0x72 0x26 }

获取配置失败：

LOG\_CMD\_ACK:{

"ut":946710768.443,"cpname":"sprs","cmd":"GET\_EQ\_GET\_SDAT",

"status":"err","err\_code":"0xd2"

}

## 自定义指令

命令格式同上，指令表暂未添加。

输入：test sprs self\_def#addr

示例：test sprs self\_def#1

说明：通常用于获取子设备的采样数据，SPRS会根据指令表中的设备指令类型（单、双指令），自动执行单/双指令数据获取，地址输入范围1-247。

程序检查项：

1：指令表是否存在；

2：读出的指令块大小是否为16；

3：所查询的设备地址不为0；

4：指令表打开正常；

5：读取的指令块大小正常；

6：子设备超时未响应。

7：子设备返回的数据中，返回的地址不对或CRC校验不对。

## 透传指令

输入：test sprs pass#xx#xx#xx…

示例：test sprs pass#01#04#00#10#00#13#b0#02

说明：SPRS在执行透传指令时，不会更换地址、也不回重新计算CRC，而是将收到的透传数据直接发送到总线上，对于接受的数据也是一样，不再进行校验检查，参数输入时是以2位16进制为一个数据的，支持大写、小写和数字。参数输入错误，会导致错误的访问。

成功：

LOG\_CMD\_ACK:{

"ut":947144218.517,"cpname":"sprs","cmd":"SEN\_EQ\_PAS",

"status":"suc","err\_code":"0x00"

}

LOG\_CMD\_RES:{

"ut":947144218.527,"cpname":"sprs","cmd":"SEN\_EQ\_PAS",

"value":0x01 0x04 0x26 0x00 0x06 0xac 0x5c 0x01 0x22 0x41 0xc5 0x80 0x00 0x3e 0xa8 0x1a 0xab 0x43 0x03 0x54 0xd6 0x41 0x47 0x6d 0xbb 0x3e 0x90 0x74 0x80 0x41 0x47 0x6d 0xbb 0x3e 0x8d 0xeb 0xc0 0x41 0xa0 0xc1 0x4e 0x49 0x0d }

失败：

LOG\_CMD\_ACK:{

"ut":946710854.133,"cpname":"sprs","cmd":"SEN\_EQ\_PAS",

"status":"err","err\_code":"0xd2"

}

说明：测试组件针对HCF710进行了数据解析。