

SDM-API使用说明书

——设备管理

V3.4.2

版权所有 盗版必究

江苏紫清信息科技有限公司

更新日期：2017 年 11 月 12 日

江苏紫清信息科技有限公司

说明

1. 本文档拟制部门：江苏紫清信息科技有限公司
 2. API 当前部署地址为：<http://s1.anylinkcloud.com:8600>，简称为
<%=BaseUrl%>
 3. 除非特殊说明，本文档中的接收和返回参数编码方式都为 UTF-8
 4. 调用所有接口（除登录接口外）前，需要先调用/user/login 接口获取 token 值
 5. token 当前有效期为 600 分钟。
-

目 录

1 接口文档.....	3
1.1 用户管理.....	3
1.1.1 用户登录.....	3
1.1.2 多租户方式获取 token.....	4
1.1.3 修改密码.....	5
1.2 设备管理.....	6
1.2.1 获取租户内授权的网关列表.....	6
1.2.2 获取网关列表.....	7
1.2.3 获取设备列表.....	9
1.2.4 获取单个设备信息.....	11
1.2.5 获取设备模型列表.....	13
1.2.6 获取某个设备的所有数据项.....	14
1.2.7 获取 device id.....	16
1.3 告警规则.....	17
1.3.1 根据设备 ID 获取配置的规则.....	17
1.4 告警.....	22
1.4.1 获取设备的告警信息.....	22
1.4.2 根据设备 ID、数据项 name 获取历史告警.....	24
1.5 实时数据.....	28
1.5.1 获取设备的实时数据.....	28
1.5.2 获取多个设备的实时数据.....	30
1.6 历史数据.....	32
1.6.1 获取单个数据项的历史数据.....	32
1.6.2 根据数据项 id 获取历史数据.....	34
1.6.3 根据数据项名称获取历史数据.....	36
1.6.4 根据数据项 id 获取历史数据(5000).....	39
1.6.5 根据数据项名称获取历史数据(5000).....	39
1.7 设备控制.....	40

1.7.1 向设备发送命令.....	40
1.7.2 获取操作结果.....	41
1.8 网关相关接口.....	42
1.8.1 获取网关的在线状态.....	42
2 错误码列表.....	44

1 接口文档

1.1 用户管理

1.1.1 用户登录

功能：用户登录，并返回 token 信息

请求方式：POST

url: </user/login>

请求参数：

格式：json

参数说明：

参数	取值	是否必需	说明
name	String	是	用户名
password	String	是	密码
hash	String	是	哈希值

返回参数：

格式：json

参数说明：

参数	取值	说明
status	String	请求状态。成功：100；失败：其它
msg	String	错误信息（status 为其它时）
data	String	Token 值

请求示例：

```
{
  name:"user1",
  password:"123456",
  hash:"test"
```

```
}
```

返回示例:

```
{
  status: "100",
  token: "dc91c34b-6a11-437a-ab8e-fa6fdc3e547b"
}
```

1.1.2 多租户方式获取 token

功能: 多租户方式获取 token

请求方式: POST

url: </user/getToken>

请求参数:

格式: json

参数说明:

参数	取值	是否必需	说明
tenantEname	String	是	租户英文名
name	String	是	租户内的用户名
password	String	是	密码
hash	String	是	哈希值

返回参数:

格式: json

参数说明:

参数	取值	说明
status	String	请求状态。成功: 100; 失败: 其它
msg	String	错误信息 (status 为其它时)
data	String	Token 值

请求示例:

```
{
  tenantEname:"test_tenant",
  name:"user1",
  password:"123456",
  hash:"test"
}
```

返回示例:

```
{
  status: "100",
  token: "dc91c34b-6a11-437a-ab8e-fa6fdc3e547b"
}
```

1.1.3 修改密码

功能: 修改密码

请求方式: PUT

url: </user/password>

请求参数:

格式: json

参数说明:

参数	取值	是否必需	说明
token	String	是	通行证
hash	String	是	哈希值
userid	Integer	是	用户 id
password	String	是	新密码

返回参数:

格式: json

参数说明:

参数	取值	说明
status	String	请求状态。成功: 100; 失败: 其它

msg	String	错误信息（status 为其它时）
-----	--------	-------------------

请求示例：

<http://<%=BaseUrl%>/user/password?token=dc91c34b-6a11-437a-ab8e-fa6fdc3e547b&hash=test&userid=102&password=234567>

返回示例：

```
{
  status:"100"
}
```

1.2 设备管理

1.2.1 获取租户内授权的网关列表

功能：该接口用于获取某个租户下的网关列表，租户内不同权限的用户获取的数据结果集不同。租户内的系统管理员用户可以获取该租户内的所有网关列表。

请求方式：GET

url：[/agentList](#)

请求参数：

参数	取值	是否必需	说明
token	String	是	用户通行证 获取方式参照【1.1.2 多租户方式获取 token】

返回参数：

格式：json

参数说明：

参数	取值	说明
status	String	请求状态。成功：100；失败：其它

msg	String	错误信息（status 为其它时）
data	JSON	结果集

```
{
  "status": "100",
  "data": [
    {
      "serialNumber": /* 网关序列号 */
      "agentCondition": 1, /* 网关在线状态：0--离线，1--在线 */
      "protocol": "http", /* 网关通信协议，http,mqtt 等 */
      "deviceList": [
        {
          "condition": 1, /* 设备的在线状态，0--在线，1--离线 */
          "deviceId": 1229811922, /* 设备 id */
          "deviceName": "DA500 测试", /* 设备名称 */
          "serialNumber": "1200988" /* 网关序列号 */
        }
      ]
    },
    .....
  ]
}
```

1.2.2 获取网关列表

功能：该接口用于获取网关列表。

请求方式：GET

url: </agentList/sys>

请求参数:

参数	取值	是否必需	说明
token	String	是	用户通行证
hash	String	是	

返回参数:

格式: json

参数说明:

参数	取值	说明
status	String	请求状态。成功: 100; 失败: 其它
msg	String	错误信息 (status 为其它时)
data	JSONArray	网关列表
agentCondition	int	网关在线状态 1-在线 0-离线
protocol	String	网关通信协议, http, mqtt 等
serialNumber	int	网关序列号
deviceList	JSONArray	设备列表
condition	int	设备的在线状态, 如果该设备长时间没有数据上传, 认为设备离线
deviceId	int	设备 ID
deviceName	String	设备名称
serialNumber	String	网关序列号

```
{
  "status": "100",
  "data": [
```

```
{
  "serialNumber": /* 网关序列号 */
  "agentCondition": 1, /* 网关在线状态: 0--离线, 1--在线 */
  "protocol": "http", /* 网关通信协议, http,mqtt 等 */
  "deviceList":[
    {
      "condition": 1, /* 设备的在线状态, 0--在线, 1--离线 */
      "deviceId": 1229811922, /* 设备 id */
      "deviceName":"DA500 测试", /* 设备名称 */
      "serialNumber":"1200988" /* 网关序列号 */
    }
  ]
},
.....
]
```

1.2.3 获取设备列表

功能: 获取设备列表

请求方式: GET

url: </devicelist>

请求参数:

参数	取值	是否必需	说明
token	String	是	用户通行证
hash	String	是	哈希值
name	String	否	设备名（模糊查询）

modelid	Integer	否	类型 id
serialnumber	String	否	序列号

返回参数:

格式: json

参数说明:

参数	取值	说明
status	String	请求状态。成功: 100; 失败: 其它
msg	String	错误信息 (status 为其它时)
data	array	设备信息列表
id	String	设备 id
name	String	设备名
serialNumber	String	序列号
lastContact	Date	上次联系时间
lastRegister	Date	注册时间
pingRate	Integer	Ping 频率
description	String	描述
deviceAttributes	String	设备属性描述字符串
deviceCondition	int	0-设备离线, 1-设备在线
channel	int	Anylink-SE 的通道 ID
channelName	String	Anylink-SE 的通道名称
deviceModel	Object	设备类型
id	Integer	类型 id
name	String	类型名
alias	String	类型别名
manufacturer	String	厂商
description	String	类型描述

请求示例:

<=<BaseUrl%>/devicelist?token=8029cd9c-257c-4f3b-9347-f7acbfa44498&hash=test>

返回示例:

```
{
  status: "100",
  data: [
    {
      "id":2,
      "name":"mini",
      "serialNumber":"3",
      "deviceAttributes":null,
      "lastContact":"2017-06-08 23:39:50.341",
      "lastRegister":"2017-06-08 23:39:50.341",
      "pingRate":0,
      "deviceCondition":0,
      "description":null,
      "channel":1,
      "channelName":"CHANNEL-1",
      "deviceModel":{
        "id":8,
        "name":"mini",
        "alias":null,
        "manufacturer":null,
        "description":null
      }
    },
    ...
  ]
}
```

1.2.4 获取单个设备信息

功能: 获取设备信息

请求方式: GET

url: </device/{dev id}>

请求参数:

参数	取值	说明
token	String	用户通行证

hash	String	哈希值
------	--------	-----

返回参数:

格式: json

参数说明:

参数	取值	说明
status	String	请求状态。成功: 100; 失败: 其它
msg	String	错误信息 (status 为其它时)
data	array	设备信息列表
同上

请求示例:

<http://localhost:8080/anylink/device/1000001?token=8029cd9c-257c-4f3b-9347-f7acbfa44498>

返回示例:

```
{
  status:"100",
  data: {
    "id":2,
    "name":"mini",
    "serialNumber":"3",
    "deviceAttributes":null,
    "lastContact":"2017-06-08 23:39:50.341",
    "lastRegister":"2017-06-08 23:39:50.341",
    "pingRate":0,
    "deviceCondition":0,
    "description":null,
    "channel":1,
    "channelName":"CHANNEL-1",
    "deviceModel":{
      "id":8,
      "name":"mini",
      "alias":null,
      "manufacturer":null,
      "description":null
    }
  }
}
```

```
}
```

1.2.5 获取设备模型列表

功能：获取设备类型类别

请求方式：GET

url：</devicemodelist>

请求参数：

参数	取值	说明
token	String	用户通行证
hash	String	哈希值

返回参数：

格式：json

参数说明：

参数	取值	说明
status	String	请求状态。成功：100；失败：其它
msg	String	错误信息（status 为其它时）
data	array	设备数据项
Id	Integer	类型 id
name	String	类型名
alias	String	类型别名
description	String	类型描述

请求示例：

<=<BaseUrl%>/devicemodelist?token=8029cd9c-257c-4f3b-9347-f7acbfa44498&hash=test>

返回示例：

```
{
  status: "100",
  data: [
    {
      id: 2,
```

```
        name: "zigbee"
      },
      {
        id: 3,
        name: "OMRON-C"
      }
    ]
  }
}
```

1.2.6 获取某个设备的所有数据项

功能： 获取设备类型类别

请求方式： GET

url: </dataitemlist/getDataItemList>

请求参数：

参数	取值	是否必需	说明
token	String	是	用户通行证
hash	String	是	哈希值
deviceid	Integer	是	设备 ID

返回参数：

格式： json

参数说明：

参数	取值	说明
status	String	请求状态。成功：100；失败：其它
msg	String	错误信息（status 为其它时）
data	array	设备数据项
devid	Integer	设备 id
itemid	Integer	数据项 ID
config	String	配置信息
datatype	String	数据类型
driverid	Integer	驱动 ID
driverName	String	驱动名称

did	Integer	
itemname	String	数据项名称
itemalias	String	数据项别名
readonly	int	0-只读 1-可写
frequency	int	数据项采集频率（毫秒）

请求示例：

<http://<%=BaseUrl%>/dataitemlist/getDataItemList?token=08a67a16-9e96-4bfd-8492-3f415585639c&hash=test&deviceid=3000537>

返回示例：

```
{
  "status": "100",
  "data": [
    {
      "devid":1433617410,
      "itemid":"2",
      "config":"1;3;4100;1;WORD;0;0;0",
      "datatype":"a",
      "driverid":4,
      "driverName":"libDModbus",
      "did":2,
      "itemname":"D4",
      "itemalias":"D4",
      "readonly":0,
      "frequency":10000
    },
    {
      "devid":1433617410,
      "itemid":"3",
```

```
        "config": "1;3;4111;1;WORD;0;0;0",
        "datatype": "a",
        "driverid": 4,
        "driverName": "libDModbus",
        "did": 2,
        "itemname": "D15",
        "itemalias": "D15",
        "readonly": 0,
        "frequency": 10000
    }
]
}
```

1.2.7 获取 device id

功能： 获取设备类型类别。如果参数中的 deviceName, agentID 都为空，则获取该用户有权限查看的所有设备 ID；如果 deviceName 为空，则获取 agentID 该序列号下所有设备 ID；如果 agentID 为空，则获取所有 deviceName 相同的设备 ID。

请求方式： GET

url: </devicelist/getDeviceID>

请求参数：

参数	取值	是否必需	说明
token	String	是	用户通行证
hash	String	是	哈希值
deviceName	String	否	设备名称
agentID	Integer	否	agent 序列号

返回参数：

格式： json

参数说明：

参数	取值	说明
----	----	----

status	String	请求状态。成功：100；失败：其它
msg	String	错误信息（status 为其它时）
data	Integer	device id

请求示例：

<http://<%=BaseUrl%>/devicelist/getDeviceID?token=08a67a16-9e96-4bfd-8492-3f415585639c&hash=test&deviceName=设备 3&agentID=1200001>

返回示例：

```
{
  "status": "100",
  "data": 1228801025,
}
```

1.3 告警规则

1.3.1 根据设备 ID 获取配置的规则

功能：根据设备 ID 获取规则

请求方式：GET

url: <http://<%=BaseUrl%>/rule/getRuleByDataItemName>

请求参数：

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
devid	String	是	设备 id
itemNames	String	否	数据项名称。 各个数据项名称之间用逗号分隔。 如果数据项名称为空，则是搜

			索该设备的所有规则。
--	--	--	------------

返回参数:

格式: json

参数说明:

参数				取值	说明
status				String	请求状态。成功：100；失败：其它
msg				String	错误信息（status 为其它时）
data	devid			String	设备 ID
	ruleID			String	规则 ID
	ruleName			String	规则名称
	triggerPO	triggerId		String	触发器 ID
		triggerName		String	触发器名称
		relation		String	各个规则之间的关系，比如“与”“或”
		type		String	
		trgItemList	itemId	String	数据项 ID
			itemName	String	数据项名称
			operand	String	阈值
			operator	String	操作符，比如>，<，=，≥，≤，≠
			depend	String	
	actionList	actionId		String	触发的动作 ID

		description		String	告警描述
		name		String	告警名称
		severity		String	告警级别，目前 包 括 8,9,10,11,12 五个级别
		type		String	
		typeName		String	

请求示例：

<http://<%=BaseUrl%>/rule/getRuleByDataItemName?token=4d38abea-5f3e-4c9b-9ea-c-9354dala254d&devid=1000253&itemNames=X1,X2>

返回示例：

```
{
  "status": "100",
  "data": [
    {
      "actionList": [
        {
          "actionId": 9,
          "description": "测试,$0",
          "name": "X1",
          "severity": "8",
          "type": "alarm 告警",
          "typeName": "alarm"
        }
      ],
      "devid": 1000253,
      "ruleID": 3,
      "ruleName": "测试",
    }
  ]
}
```

```
"triggerPO": {
  "relation": "0",
  "trgItemList": [
    {
      "depend": "",
      "itemId": 0,
      "itemName": "X1",
      "operand": "111",
      "operator": "="
    }
  ],
  "triggerId": 17,
  "triggerName": "测试",
  "type": "1"
},
{
  "actionList": [
    {
      "actionId": 10,
      "description": "111,$0",
      "name": "X1",
      "severity": "8",
      "type": "alarm 告警",
      "typeName": "alarm"
    }
  ],
  "devid": 1000253,
  "ruleID": 4,
  "ruleName": "111",
```

```
"triggerPO": {
  "relation": "0&1",
  "trgItemList": [
    {
      "depend": "",
      "itemId": 0,
      "itemName": "X1",
      "operand": "-1",
      "operator": ">"
    },
    {
      "depend": "",
      "itemId": 1,
      "itemName": "X1",
      "operand": "1",
      "operator": "≤"
    }
  ],
  "triggerId": 18,
  "triggerName": "111",
  "type": "1"
}
]
}
```

1.4 告警

1.4.1 获取设备的告警信息

功能：获取设备的告警信息

请求方式：GET

url: </alarm>

请求参数：

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
devid	String	是	设备 id
severity	String	否	告警级别。 取值：8，9，10，11，12 分别对应：一级，二级，三级，四级，通知
stime	Long	否	开始时间 (时间戳，单位毫秒)
etime	Long	否	结束时间 (时间戳，单位毫秒)
pageindex	int	否	第几页，用于数据分页，默认为 1
pagesize	int	否	每页显示的行数，默认 20
orderby	String	否	按照某个字段排序，默认按照 htime 排序
desc	Boolean	否	可选值：true, false 是否按照倒序排序。默认是 true。

返回参数：

格式：json

参数说明：

参数	取值	说明
status	String	请求状态。成功：100；失

				败：其它
msg			String	错误信息（status 为其它时）
data	list	ack	String	
		active	String	
		alarmdesc	String	告警描述
		alarmid	String	告警 ID
		alarmname	String	告警名称
		auxid	String	
		dataitemid	String	
		dataitemvalue	String	
		deschashcode	String	
		deviceid	String	设备 ID
		htime	Long	触发告警的时间（毫秒）
		happentime	Long	服务器接收到数据的时间（毫秒）
		notesupdateddate	String	
		priority	String	
		ruleid	String	告警规则 ID
		severity	String	告警级别，取值为 8,9,10,11,12
		thisDataItemName	String	触发该告警的数据项名称
		thisDataItemValue	String	触发告警时该数据项的值
	totalSize		String	此次查询数据条数

请求示例：

<http://<%=BaseUrl%>/alarm?token=77490586-17dc-4add-9a1c-7735029ac56e&devid=1000253&severity=8&pageindex=1&pagesize=20&orderby=htime&desc=true>

返回示例：

{

```

    "status": "100",
    "data": {
      "list": [
        {
          "ack": false,
          "active": true,
          "alarmdesc": "烟雾告警",
          "alarmid": 6,
          "alarmname": "防区 5",
          "auxid": "防区 5",
          "dataitemid": 1,
          "dataitemvalue": " ",
          "deschashcode": 0,
          "deviceid": 1536115713,
          "happentime": 1486956769561,
          "hetime": 1486956769528,
          "notesupdateddate": 0,
          "priority": 1,
          "ruleid": 1,
          "severity": 11,
          "thisDataItemName": "NAME2",
          "thisDataItemValue": ""
        }
      ],
      "totalSize": 1
    }
  }

```

1.4.2 根据设备 ID、数据项 name 获取历史告警

功能： 获取单个数据项的历史数据

请求方式： GET

url: </hisAlarm>

请求参数：

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
devid	String	是	设备 id

itemName	String	否	数据项名称，各名称之间用逗号分隔
severity	String	否	告警级别。 取值：8，9，10，11，12 分别对应：一级，二级，三级，四级，通知
stime	Long	否	开始时间(时间戳，单位毫秒)
etime	Long	否	结束时间(时间戳，单位毫秒)
pageIndex	int	否	第几页，用于数据分页，默认取值为1
pagesize	int	否	每页显示的行数，默认20
orderby	String	否	按照某个字段排序，默认字段为htime
desc	Boolean	否	可选值：true，false 是否按照倒序排序。默认是true。

返回参数：

格式： json

参数说明：

参数			取值	说明
status			String	请求状态。成功：100；失败：其它
msg			String	错误信息(status为其它时)
data	list	ack	String	
		active	String	
		alarmdesc	String	告警描述
		alarmid	String	告警 ID

	alarmname	String	告警名称
	auxid	String	
	dataitemid	String	
	dataitemvalue	String	
	deschashcode	String	
	deviceid	String	设备 ID
	htime	Long	触发告警的时间 (毫秒)
	happentime	Long	服务器接收到数据 的时间 (毫秒)
	notesupdateddate	String	
	priority	String	
	ruleid	String	告警规则 ID
	severity	String	告警级别
	thisDataItemName	String	触发该告警的数据 项名称
	thisDataItemValue	String	触发告警时该数据 项的值
totalSize		String	总的告警条数

请求示例:

[http://<%=BaseUrl%>/hisAlarm?token=77490586-17dc-4add-9a1c-7735029ac56e&dev
id=1000253&itemNames=X1&severity=8&pageindex=1&pagesize=3&orderby=htim
e&desc=true](http://<%=BaseUrl%>/hisAlarm?token=77490586-17dc-4add-9a1c-7735029ac56e&dev
id=1000253&itemNames=X1&severity=8&pageindex=1&pagesize=3&orderby=htim
e&desc=true)

返回示例:

```
{
  "status": "100",
  "data": {
    "list": [
      {
        "ack": false,
        "active": true,
        "alarmdesc": "测试",
```

```

        "alarmid": 375,
        "alarmname": "alarm",
        "auxid": "",
        "dataitemid": 0,
        "dataitemvalue": " ",
        "deschashcode": 0,
        "happentime": 1432865815224,
        "htime": 1432865787034,
        "notesupdateddate": 0,
        "priority": 1,
        "ruleid": 3,
        "severity": 8,
        "thisDataItemName": "X1",
        "thisDataItemValue": "0"
    },
    {
        "ack": false,
        "active": true,
        "alarmdesc": "测试",
        "alarmid": 374,
        "alarmname": "alarm",
        "auxid": "",
        "dataitemid": 0,
        "dataitemvalue": " ",
        "deschashcode": 0,
        "happentime": 1432865815203,
        "htime": 1432865786033,
        "notesupdateddate": 0,
        "priority": 1,
        "ruleid": 3,
        "severity": 8,
        "thisDataItemName": "X1",
        "thisDataItemValue": "0"
    },
    {
        "ack": false,
        "active": true,
        "alarmdesc": "测试",
        "alarmid": 373,
        "alarmname": "alarm",
        "auxid": "",
        "dataitemid": 0,
        "dataitemvalue": " ",
        "deschashcode": 0,

```

```
        "happentime": 1432865813208,
        "hetime": 1432865785033,
        "notesupdateddate": 0,
        "priority": 1,
        "ruleid": 3,
        "severity": 8,
        "thisDataItemName": "X1",
        "thisDataItemValue": "0"
    }
],
"totalSize": 185
}
```

1.5 实时数据

1.5.1 获取设备的实时数据

功能： 获取实时数据

请求方式： GET

url: [/currentdata](#)

请求参数：

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
deviceid	String	是	设备 id
itemids	String	否	要查询的 itemid 列表, 多个时以 “,” 隔开
itemnames	String	否	要查询的 itemname 列表, 多个时以 “,” 隔开。 itemids 条件存在且不为空时, 此项将被忽略。

返回参数:

格式: json

参数说明:

参数	取值	说明
status	String	请求状态。成功: 100; 失败: 其它
msg	String	错误信息(status 为其它时)
data	array	设备数据项
itemid	String	数据项 id
devid	String	设备 id
itemname	String	数据项名
alias	String	别名
datatype	String	数据类型
val	Object	数据项值
readOnly	bool	ture-只读 false-可写
htime	String	时间
dataAddress	String	数据点地址
devName	String	设备名称
quality	String	g--good, b--bad
specificType	String	功能码

请求示例:

[<%=BaseUrl%>/currentdata?token=c5155039-ef0d-45b9-8b43-5c73cf1b9c6f&hash=test&deviceid=1000002&itemids=1,2,3](#)

返回示例:

```
{
  "status": "100",
  "data": [
    {
      "alias": "NAME1",
      "dataAddress": "0",
```

```

        "datatype": "3",
        "devName": "DEVICE1",
        "devId": "1536006145",
        "hTime": "2017-01-21 18:19:20.0",
        "itemId": "1",
        "itemName": "NAME1",
        "quality": "b",
        "readOnly": false,
        "specificType": "3",
        "val": "6.0"
    },
    ...
]
}

```

1.5.2 获取多个设备的实时数据

功能： 获取多个设备的实时数据

请求方式： GET

url: </currentdata/getDeviceCurrentData>

请求参数：

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
deviceIds	String	是	设备 id，多项之间用英文逗号相隔

返回参数：

格式： json

参数说明：

参数	取值	说明
status	String	请求状态。成功：100；失败：其它
msg	String	错误信息(status 为其它时)

data	array	设备数据项
itemid	String	数据项 id
devid	String	设备 id
itemname	String	数据项名
alias	String	别名
datatype	String	数据类型
val	Object	数据项值
readOnly	bool	ture-只读 false-可写
htime	String	时间
dataAddress	String	数据点地址
devName	String	设备名称
quality	String	g--good, b--bad
specificType	String	功能码

请求示例:

localhost:8080/anylink/currentdata/getDeviceCurrentData?token=c5155039-ef0d-45b9-8b43-5c73cf1b9c6f&hash=test&deviceIds=1228801025,1228801026

返回示例:

```
{
  "status": "100",
  "data": [
    {
      "alias": "NAME1",
      "dataAddress": "0",
      "datatype": "3",
      "devName": "DEVICE1",
      "devid": "1228801025",
      "htime": "2017-01-21 18:19:20.0",
      "itemid": "1",
      "itemname": "NAME1",
      "quality": "g",
      "readOnly": false,
      "specificType": "3",
      "val": "6.0"
    },
    ...
  ]
}
```

```
]
}
```

1.6 历史数据

1.6.1 获取单个数据项的历史数据

功能： 获取单个数据项的历史数据，数据按照 **htime** 降序排列，获取最新的数据。

请求方式： GET

url: </historydata>

请求参数：

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
deviceid	Integer	是	设备 id
dataitemid	String	是	数据项 id
stime	Long	否	开始时间，时间戳，单位毫秒
etime	Long	否	结束时间，时间戳，单位毫秒
limit	Integer	否	条数限制（默认 1000 条，最多支持 1000 条）
format	String	否	返回格式(json/file) file 文件为 excel

返回参数：

格式： json

参数说明：

参数	取值	说明
----	----	----

status	String	请求状态。成功: 100; 失败: 其它
msg	String	错误信息 (status 为其它时)
data	array	设备数据项
devid	String	设备 id
itemid	String	数据项 id
itemname	String	数据项名称
alias	String	数据项别名
val	String	数据值
htime	String	时间戳
datatype	String	数据类型
datatypeName	String	数据类型名称
readOnly	Boolean	true-只读 false-可写

请求示例:

<http://localhost:8080/historydata?token=90e1a491-2360-4eb4-90e8-e3f2379192c2&hash=test&deviceid=20&dataitemid=24&stime=1412750888000&etime=1416898091000>

返回示例:

```
{
  "status": "100",
  "data": [
    {
      "alias": "N1",
      "datatype": "1",
      "datatypeName": "Boolean",
      "devid": "1536008193",
      "htime": "2017-03-07 10:11:27.166",
      "itemid": "1",
      "itemname": "N1",
      "readOnly": true,
      "val": "1"
    }
  ],
}
```

```
{
  {
    "alias": "N1",
    "datatype": "1",
    "datatypeName": "Boolean",
    "devid": "1536008193",
    "htime": "2017-03-07 10:11:26.291",
    "itemid": "1",
    "itemname": "N1",
    "readOnly": true,
    "val": "1"
  },
  {
    "alias": "N1",
    "datatype": "1",
    "datatypeName": "Boolean",
    "devid": "1536008193",
    "htime": "2017-03-07 10:11:25.303",
    "itemid": "1",
    "itemname": "N1",
    "readOnly": true,
    "val": "1"
  }
}
```

1.6.2 根据数据项 id 获取历史数据

功能：根据数据项 id（可以是多个）获取历史数据

请求方式：GET

url: </history/forids>

请求参数：

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
deviceid	String	是	设备 id
itemids	String	否	数据项 id 列表，多个时以

			“,” 隔开。不传时默认查询所有数据项历史数据
stime	Long	否	开始时间, 时间戳, 单位毫秒
etime	Long	否	结束时间, 时间戳, 单位毫秒
limit	Integer	否	每个数据项数据的条数限制, 最大 1000 条, 所有数据项总条数最大为 100000 条, 如果查询数据项个数 $n \times 1000 > 100000$, 那么每个数据项数据的最大条数为 $100000/n$
format	String	否	返回格式 (json/file)

返回参数:

格式: json

参数说明:

参数		取值	说明
status		String	请求状态。成功: 100; 失败: 其它
msg		String	错误信息 (status 为其它时)
data		Object	内容格式为: itemid: 该数据项的历史数据列表
itemid	devid	String	设备 id
	itemid	String	数据项 id
	itemname	String	数据项名称
	alias	String	数据项别名
	val	String	数据值
	htime	String	时间戳
	datatype	String	数据类型

	datatypeName	String	数据类型名称
	readOnly	Boolean	true-只读 false-可写

请求示例: [<%=BaseUrl%>/history/](#)

[forids?token=a0d01cf4-6f78-4f8a-8db0-81bb35978196&has
h=test&deviceid=12&itemids=11,12](#)

返回示例:

```
{
  "status": "100",
  "data": {
    "1": [
      {
        "alias": "N1",
        "datatype": "1",
        "datatypeName": "Boolean",
        "devid": "1536008193",
        "htime": "2017-03-07 10:11:27.166",
        "itemid": "1",
        "itemname": "N1",
        "readOnly": true,
        "val": "1"
      },
      {
        "alias": "N1",
        "datatype": "1",
        "datatypeName": "Boolean",
        "devid": "1536008193",
        "htime": "2017-03-07 10:11:26.291",
        "itemid": "1",
        "itemname": "N1",
        "readOnly": true,
        "val": "1"
      },
      ...
    ]
  }
}
```

1.6.3 根据数据项名称获取历史数据

功能: 根据数据项名称 (可以是多个) 获取历史数据

请求方式: GET

url: </history/fornames>

请求参数:

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
deviceid	String	是	设备 id
itemnames	String	否	数据项名称列表, 多个时以 “,” 隔开。不传时默认查询所有数据项历史数据
stime	Long	否	开始时间, 时间戳, 单位毫秒
etime	Long	否	结束时间, 时间戳, 单位毫秒
limit	Int	否	每个数据项数据的条数限制, 最大 1000 条, 所有数据项总条数最大为 100000 条, 如果查询数据项个数 $n*1000 > 100000$, 那么每个数据项数据的最大条数为 $100000/n$
format	String	否	返回格式 (json/file)

返回参数:

格式: json

参数说明:

参数	取值	说明
status	String	请求状态。成功: 100; 失败: 其它
msg	String	错误信息 (status 为其它时)

data		Object	内容格式为: itemid:该数据项的历史数据列表
itemid	devid	String	设备 id
	itemid	String	数据项 id
	itemname	String	数据项名称
	alias	String	数据项别名
	val	String	数据值
	htime	String	时间戳
	datatype	String	数据类型
	datatypeName	String	数据类型名称
	readOnly	Boolean	true-只读 false-可写

请求示例:

[<%=BaseUrl%>/history/fornames?token=a0d01cf4-6f78-4f8a-8db0-81bb35978196&hash=test&deviceid=12&itemnames=data11,data12](http://localhost:8080/history/fornames?token=a0d01cf4-6f78-4f8a-8db0-81bb35978196&hash=test&deviceid=12&itemnames=data11,data12)

返回示例:

```

{
  "status": "100",
  "data": {
    "data11": [
      {
        "datatype": "b",
        "datatypeName": "开关量",
        "devid": "12",
        "htime": "2014-12-01 23:59:56",
        "itemid": "11",
        "itemname": "data11",
        "readOnly": true,
        "val": "0"
      },
      ...
    ],
    "data12": [
      {
        "datatype": "b",

```



```
        "datatypeName": "开关量",
        "devid": "12",
        "htime": "2014-12-01 23:59:56",
        "itemid": "12",
        "itemname": "data12",
        "readOnly": true,
        "val": "0"
    },
    ...
]
```

1.6.4 根据数据项 id 获取历史数据(5000)

版本: v2.1

功能: 根据数据项 id (可以是多个) 获取历史数据

请求方式: GET

url: </history/foridsNoLimit>

其余: 同 1.9.2

和 1.9.2 不同在于: 每个数据项获取的历史数据最多为 5000 条。

1.6.5 根据数据项名称获取历史数据(5000)

版本: v2.1

功能: 根据数据项名称 (可以是多个) 获取历史数据

请求方式: GET

url: </history/fornamesNoLimit>

其余: 同 1.9.2

和 1.9.2 不同在于: 每个数据项获取的历史数据最多为 5000 条。

1.7 设备控制

1.7.1 向设备发送命令

版本： V2.1

功能： 向设备发送命令（只返回发送是否成功，不返回操作结果），改变数据项的值

请求方式： PUT

url: </control/v2.1>

请求参数：

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
devid	int	是	设备 id
driverID	int	否	驱动 id
driverName	String	否	驱动名称 如果为空：如果 itemid!=null,则获取该 item 对应的驱动名称； 如果 itemid==null，该值默认为"libTCmd"； DA RA 切换时，itemid=null，并且 driverName=libtremotepkc
itemid	int	否	数据项 id，DA RA 切换时，itemid=null
itemname	String	否	数据项名称，itemid 和 itemname 必须至少有一个不为空，当 itemid 存在且不为空时，itemname 会被忽略
value	String	是	数据项的值 (开关量类型的数据项值为 0 或 1)

返回参数:

格式: json

参数说明:

参数	取值	说明
status	String	请求状态。 100: 控制命令成功加入队列; 103: 参数错误 106: 其他用户控制中 111: 未知错误
msg	String	错误信息 (status 为其它时)
data	String	用来获取控制结果的标识码

请求示例:

[<%=BaseUrl%>/control?token=90e1a491-2360-4eb4-90e8-e3f2379192c2&hash=test&devid=1001&itemid=48&value=1](#)

返回示例:

```
{  
  status:"100",  
  data: 1415844000000  
}
```

1.7.2 获取操作结果

功能: 获取操作结果 (需循环访问, 直到得到结果)

请求方式: GET

url: [/control/result](#)

请求参数:

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
devid	String	是	设备 id

sign	String	是	用于获取结果的标识，取值为 /control/v2.1 接口中获取的 data 值
------	--------	---	---

返回参数：

格式： json

参数说明：

参数	取值	说明
status	String	请求状态。成功：100；失败：其它
msg	String	错误信息（status 为其它时）
data	String	结果（0：成功，3：超时，其他：失败）

请求示例：

[<%=BaseUrl%>/control/result?token=90e1a491-2360-4eb4-90e8-e3f2379192c2&hash=test&devid=1002&itemid=48&sign=1415844000000](#)

返回示例：

```
{
  status:"100",
  data:"1"
}
```

1.8 网关相关接口

1.8.1 获取网关的在线状态

功能： 根据网关序列号，获取该网关配套安装的 SIM 卡号

请求方式： POST

url: [/agentList/condition](#)

请求参数：

参数	取值	是否必须	说明
token	String	是	用户通行证
hash	String	是	哈希值
agentIds	String	是	网关序列号列表

请求参数示例：

```
{
  "agentIds": [
    1500001,
    1500002
  ],
  "token": "8534f960-8471-4dec-b470-cda3d348ecbc"
}
```

返回参数：

格式： json

参数说明：

参数	取值	说明
status	String	请求状态。成功：100；失败：其它
msg	String	错误信息（status 为其它时）
data	String	JSONArray
agentid	String	网关序列号
condition	String	网关的在线状态。 1---在线 0---离线

请求示例：

`<%=BaseUrl%>/agentList/condition`

返回示例：

```
{
  "status": "100",
  "data": [
    {
      "agentid": 1500001,
      "condition": 0
    }
  ]
}
```

```
    },
    {
        "agentid":1500002,
        "condition":0
    }
]
}
```

2 错误码列表

序号	错误代码	说明
1	100	成功
2	101	权限错误
3	102	超时
4	103	参数错误
5	104	无效的 token
6	106	设备被其他用户控制
7	107	用户名或密码错误
8	108	没有这个设备
9	109	没有这个用户
10	110	用户名已存在
11	111	未知错误
12	112	设备未授权
13	113	用户组名已经存在
14	114	设备组名已存在
15	115	用户已被锁定
16	116	登录密码错误
17	117	VPN 服务器已经被使用
18	118	VPN 服务不可用
19	119	设备没有 VPN 信息，比如

		没有设置网段信息
20	120	没有可用的 IP, 请重新分配 VPN 环境
21	121	绑定设备失败
22	122	设备还没有重新注册
23	123	设备重新注册失败
24	124	等待下发配置文件

了解更多信息，欢迎访问：www.anylink.io



地址：江苏省无锡市兴源北路401号北创科技园

网址：www.anylink.io

电话：400-066-9060

传真：0510-82608960

ANYLINK