

动环系统软件接口说明书

接口版本	V2.1.2
作者	
创建日期	2022.01.28

文档变更记录

序号	变更说明	作者	日期
1	创建		2022.01.28
2	修改获取实时数据接口，添加主机地址和通道号参数		2022.02.21
3	获取因子实时数据接口增加参数说明		2023.05.12
4	对接口访问频率作一定限制		2025.07.14

设计

目录

一、统一说明	1
1、通讯方式	1
2、接口调用注意事项	1
3、接口地址	1
二、接口详细	1
1、用户登录	1
2、数据操作	3
2.1 获取因子实时数据	3
2.2 获取因子历史数据	6
3、因子操作	8
3.1 获取控制因子	8
3.2 控制继电器因子	9
三、附录	10
返回代码	10

一、统一说明

1、通讯方式

数据接口采用 http 协议，通讯内容采用 JSON 数据格式。

接口返回数据格式如下：

```
{"code":1001, "message":"信息内容", "data":object}
```

ResultData 字段说明

字段	类型	说明
code	Integer	代码(参考返回代码说明表)
message	String	返回操作结果描述
data	Json	Json 数据，具体格式参考各个接口

文档中所有接口的返回数据说明表，如无特殊说明均为 ResultData 数据格式，返回的数据均存放于 data 字段中

2、接口调用注意事项

调用接口(除登录接口外)需要将 token 写入 http 请求头中。**请求头字段名固定为 token**，字段值为登录接口获取到的 token 值。

token 通过登录接口获取。根据用户名和密码调用登录接口，登录成功后接口返回 token。

接口中如无特殊说明，采用 post 方法的接口中参数为 json 格式，放入请求 body 中；采用 get 方法的接口中参数为查询参数，放入请求地址后

3、接口地址

http://ip:port（默认 80）

4、接口访问频率限制

控制继电器因子接口/sdk/FactorManage/ControlRelay 访问频率限定为 5 秒钟，其它为 2 秒钟。

二、接口详细

1、用户登录

1.1 接口描述

接口编号	1.1
接口名称	登录
功能描述	用户登录
接口地址	/sdk/UserManage/Login
请求方式	POST

参数格式	JSON
返回数据格式	String
备注	

2、数据操作

2.1 获取因子实时数据

2.1.1 接口描述

接口编号	2.1
接口名称	查询因子实时数据
功能描述	根据当前的登录用户获取其管理的项目下的因子的实时数据
接口地址	/sdk/FactorManage/GetRealTimeData
请求方式	GET
参数格式	
返回数据格式	JSON 数组
备注	

表 2.1

2.1.2 请求参数说明

请求参数

字段	类型	说明
hostAddr	Integer	主机地址
channelNo	Integer	通道号

表 2.2

请求头

字段	类型	说明
token	String	登录成功后返回的 accessToken 值

表 2.3

2.1.3 返回数据说明

字段	类型	说明
FactorId	String	因子编号
FactorName	String	因子名称
FactorType	Integer	因子类型（属性类型 0，控制类型 1，设置类型 2，平均值类型 3，M88 继电器类型 4，自定义类型 5，蓄电池组类型 6，蓄电池类型 7，蓄电池电压/温度/电阻类型 8，遥调类型 9）
HostAddr	Integer	主机地址
ChannelNo	Integer	通道号

Plugin	String	插件名称
DisplayMsg	String	因子的实时显示数据（扩展显示值，如：“非定位式水浸”因子，如果值为 1 则为闭合显示内容“报警”，如果值为 0 则为断开显示内容“正常”）
State	String	初始状态 Default，正常在线 OnLine，离线 OffLine，报警 Alarm
AlarmReset	Integer	报警复归状态（已复归 1，未复归 0）
Icon	String	根据因子的状态返回的相应的显示图标
AlarmMsg	String	报警信息（如果因子是报警状态，显示实际的报警信息）
FactorCtrlStateUpper	Integer	因子控制上限状态（初始 0，超控制上限 1，控制上限回差内 2）
FactorCtrlStateLower	Integer	因子控制下限状态（初始 0，超控制下限 1，控制下限回差内 2）
Type	Integer	标识（默认 1）
Value	String	因子的实时数据（原始值）
OrderNo	Integer	因子排序号，默认 0

表 2.4

2.1.4 返回格式示例

```
{
  "code": 1000,
  "message": "获取成功",
  "data": [
    {
      "FactorId": "2102206902001001",
      "FactorName": "1-1-RK 温度",
      "FactorType": 0,
      "HostAddr": 21022069,
      "ChannelNo": 2,
      "plugin": "Temperature",
      "DisplayMsg": "",
      "State": "OffLine",
      "AlarmReset": 0,
      "Icon": "/static/images/factor_offline.png",
      "AlarmMsg": "",
      "FactorCtrlStateUpper": 0,
      "FactorCtrlStateLower": 0,
      "Type": 1,
      "Value": "",
      "OrderNo": 0
    },
    {
      "FactorId": "2102206902002001",
      "FactorName": "2-1-RK 温度",
      "FactorType": 0,
```

```

    "HostAddr": 21022069,
    "ChannelNo": 2,
    "plugin": "Temperature",
    "DisplayMsg": "",
    "State": "OffLine",
    "AlarmReset": 0,
    "Icon": "/static/images/factor_offline.png",
    "AlarmMsg": "",
    "FactorCtrlStateUpper": 0,
    "FactorCtrlStateLower": 0,
    "Type": 1,
    "Value": "",
    "OrderNo": 0
  },
  {
    "FactorId": "2102206902001000",
    "FactorName": "1-0-RK 湿度",
    "FactorType": 0,
    "HostAddr": 21022069,
    "ChannelNo": 2,
    "plugin": "Humidity",
    "DisplayMsg": "",
    "State": "OffLine",
    "AlarmReset": 0,
    "Icon": "/static/images/factor_offline.png",
    "AlarmMsg": "",
    "FactorCtrlStateUpper": 0,
    "FactorCtrlStateLower": 0,
    "Type": 1,
    "Value": "",
    "OrderNo": 0
  },
  {
    "FactorId": "2102206902002000",
    "FactorName": "2-0-RK 湿度",
    "FactorType": 0,
    "HostAddr": 21022069,
    "ChannelNo": 2,
    "plugin": "Humidity",
    "DisplayMsg": "",
    "State": "OffLine",
    "AlarmReset": 0,
    "Icon": "/static/images/factor_offline.png",
    "AlarmMsg": "",

```

```

        "FactorCtrlStateUpper": 0,
        "FactorCtrlStateLower": 0,
        "Type": 1,
        "Value": "",
        "OrderNo": 0
    },
    {
        "FactorId": "2102206902003003",
        "FactorName": "3-3-RK 红外",
        "FactorType": 0,
        "HostAddr": 21022069,
        "ChannelNo": 2,
        "plugin": "Infrared",
        "DisplayMsg": "",
        "State": "OffLine",
        "AlarmReset": 0,
        "Icon": "/static/images/factor_offline.png",
        "AlarmMsg": "",
        "FactorCtrlStateUpper": 0,
        "FactorCtrlStateLower": 0,
        "Type": 1,
        "Value": "",
        "OrderNo": 0
    }
]
}

```

2.2 获取因子历史数据

2.2.1 接口描述

接口编号	2.2
接口名称	查询因子历史数据
功能描述	查询相应因子的历史数据
接口地址	/sdk/HistoryManage/GetData
请求方式	GET
参数格式	JSON
返回数据格式	JSON 数组
备注	

表 2.5

2.2.2 请求参数说明

请求参数

字段	类型	说明
factorName	String	因子名称（因子名称和因子 ID 必须输入其一）
factorId	String	因子 ID（因子名称和因子 ID 必须输入其一）
startTime	String	开始时间（格式：yyyy-MM-dd HH:mm:ss，如 2022-01-28 08:00:00，如果查询所有时间段则为空字符串）
endTime	String	结束时间（同上）
dataStatus	String	数据状态（正常为 0，异常为 1，全部为空字符串）

表 2.6

请求头

字段	类型	说明
token	String	登录成功后返回的 accessToken 值

表 2.7

请求 JSON:

```
{
  "factorId": "2102207002001001",
  "factorName": "",
  "startTime": "2023-01-01 00:00:00",
  "endTime": "2023-05-01 00:00:00",
  "dataStatus": "0"
}
```

2.2.3 返回数据说明

字段	类型	说明
Id	String	数据编号
DetectItemId	String	所属监测项 ID
DetectItemName	String	所属监测项名称
FactorId	String	因子 ID
FactorName	String	因子名称
Val	String	数据
SaveTime	String	保存时间
DataStatus	String	状态（正常或异常）

表 2.8

```
{
  "code": 1000,
  "message": "获取成功",
  "data":
  [
```

```
{
  "Id": "608673",
  "RoomId": null,
  "RoomName": null,
  "DetectItemId": "636F181EA9014B05B536A55924CC632E",
  "DetectItemName": "温度监测",
  "FactorId": "2102207002001001",
  "FactorName": "1-1-RK 温度",
  "Val": "20.9",
  "SaveTime": "2023-01-03T09:51:34",
  "DataStatus": "正常"
},
{
  "Id": "608683",
  "RoomId": null,
  "RoomName": null,
  "DetectItemId": "636F181EA9014B05B536A55924CC632E",
  "DetectItemName": "温度监测",
  "FactorId": "2102207002001001",
  "FactorName": "1-1-RK 温度",
  "Val": "20.9",
  "SaveTime": "2023-01-03T09:54:47",
  "DataStatus": "正常"
},
{
  "Id": "608696",
  "RoomId": null,
  "RoomName": null,
  "DetectItemId": "636F181EA9014B05B536A55924CC632E",
  "DetectItemName": "温度监测",
  "FactorId": "2102207002001001",
  "FactorName": "1-1-RK 温度",
  "Val": "20.9",
  "SaveTime": "2023-01-03T09:59:47",
  "DataStatus": "正常"
}
]
```

3、因子操作

3.1 获取控制因子

3.1.1 接口描述

接口编号	3.1
接口名称	查询所有控制因子
功能描述	
接口地址	/sdk/FactorManage/GetRelayFactors
请求方式	GET
参数格式	
返回数据格式	JSON 数组
备注	

表 3.1

3.1.2 请求参数说明

请求头

字段	类型	说明
token	String	登录成功后返回的 accessToken 值

表 3.2

3.1.3 返回数据说明

字段	类型	说明
factorId	String	因子 ID
factorName	String	因子名称
hostAddr	Integer	主机地址
channelNo	Integer	通道号
addr	Integer	设备地址

表 3.3

3.1.4 返回格式示例

```
{
  "code": 1000,
  "message": "获取成功",
  "data": [
    {
      "factorId": "7001008008",
      "factorName": "控制开关 1",
      "hostAddr": 21022070,
      "channelNo": 1,
      "addr": 8
    },
    {
      "factorId": "7001008009",
      "factorName": "控制开关 2",
      "hostAddr": 21022070,
      "channelNo": 1,
      "addr": 8
    }
  ]
}
```

3.2 控制继电器因子

3.2.1 接口描述

接口编号	3.2
接口名称	控制继电器因子
功能描述	控制指定继电器闭合/断开
接口地址	/sdk/FactorManage/ControlRelay
请求方式	POST
参数格式	POST
返回数据格式	String
备注	

表 3.4

3.2.2 请求参数说明

请求参数

字段	类型	说明
factorId	String	因子 ID
cmd	Integer	操作（闭合为 1，断开为 0）

表 3.5

请求头

字段	类型	说明
token	String	登录成功后返回的 accessToken 值

表 3.6

三、附录

返回代码

代码	说明
1000	接口执行成功，具体参考各个接口
1001	参数错误，具体参考各个接口，原因参考 Message
1002	鉴权失败
1003	程序内部异常，异常信息参考 Message
1004	操作失败