# SMDC接口使用说明

## 一.获取设备实时状态

### 1.读取单个设备变量

|  |  |
| --- | --- |
| 接口名称 | ReadTagValue  |
| 请求方式 | Get |
| 请求地址 | http://{host}/dataservice/webapi/ReadTagValue? |
| 接口说明 | 第三方软件调用该接口获取指定变量(设备参数或Tag)的值 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
| path | 字符 | 设备变量的路径 | 必输 |
| 格式举例 | path=IO.Group1.Simulator1.IOTag1 |
| 返回结果 | 类型 | JSON String |
| 格式举例 | {"Key":"IO.Group1.Simulator1.IOTag1",//设备变量的路径"Value":123,//设备变量的变量值"QualityStamp":1,//设备变量的质量戳"TimeStamp":"2019-07-03T10:31:18.2235866+08:00"//时间} |

### 2.读取多个设备变量的变量值

|  |  |
| --- | --- |
| 接口名称 | ReadTagValues |
| 请求方式 | Get |
| 请求地址 | http://{host}/dataservice/webapi/ReadTagValues? |
| 接口说明 | 第三方软件调用该接口得到的多个设备变量的变量值 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
| paths | 字符 | 设备变量的路径,以逗号隔开 | 必输 |
| 格式举例 | paths=IO.Group1.Simulator1.IOTag1,IO.Group1.Simulator1.IOTag2 |
| 返回结果 | 类型 | JSON String |
| 格式举例 | [{ "Key": "IO.Group1.Simulator1.IOTag1", "Value": 0, "QualityStamp": 1, "TimeStamp": "2019-07-03T17:57:35.5341968+08:00"}, { "Key": "IO.Group1.Simulator1.IOTag2", "Value": 0, "QualityStamp": 1, "TimeStamp": "2019-07-03T17:57:35.5341968+08:00"}] |

### 3.修改单个设备变量的值

|  |  |
| --- | --- |
| 接口名称 | WriteTagValue |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/WriteTagValue? |
| 接口说明 | 第三方软件调用该接口修改单个变量（设备参数或Tag）的值 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | {"Key":"IO.Group1.Simulator1.IOTag1","Value":123,"QualityStamp":1,"TimeStamp":"2019-07-03T10:31:18.2235866+08:00"} |
| 返回结果 | 类型 | Json 字符 |
| 格式举例 | "IO.Group1.Simulator1.IOTag1;True" |

### 4.修改多个设备变量的值

|  |  |
| --- | --- |
| 接口名称 | WriteTagValues |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/WriteTagValues? |
| 接口说明 | 第三方软件调用该接口修改多个变量（设备参数或Tag）的值 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | [{ "Key": "IO.Group1.Simulator1.IOTag1", "Value": 0, "QualityStamp": 1, "TimeStamp": "2019-07-03T17:57:35.5341968+08:00"}, { "Key": "IO.Group1.Simulator1.IOTag2", "Value": 0, "QualityStamp": 1, "TimeStamp": "2019-07-03T17:57:35.5341968+08:00"}] |
| 返回结果 | 类型 | Json 字符 |
| 格式举例 | ["IO.Group1.Simulator1.IOTag1;True","IO.Group1.Simulator1.IOTag2;True"] |

### 5.获取指定对象的属性值

|  |  |
| --- | --- |
| 接口名称 | ReadProperty |
| 请求方式 | Get |
| 请求地址 | http://{host}/dataservice/webapi/ReadProperty? |
| 接口说明 | 第三方软件调用该接口获取指定对象的属性值 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
| path | 字符串 | 设备变量的路径 | 必输 |
| propertyName | 字符串 | 属性的名字 | 必输 |
| 格式举例 | path=IO.Group1.Simulator1.IOTag1&propertyName=Name |
| 返回结果 | 类型 | Json 字符 |
| 格式举例 | { "Key": "IO.Group1.Simulator1.IOTag1.Name", "Value": "IOTag1", "QualityStamp": 32, "TimeStamp": "2019-07-04T10:57:17.3973942+08:00"} |
| 性能要求 | 响应结果3秒内返回 |

## 二.获取历史数据

### 1.获取历史数据库源数据

|  |  |
| --- | --- |
| 接口名称 | QueryToSTR |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 第三方软件调用该接口，根据设备变量的路径集合获取历史数据 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "QueryToSTR",//查询方法名 "Paras": [["IO.Simulator1.IOTag1"],//设备变量路径集合 "2020-4-23 10:47:00",//查询范围开始时间 "2020-4-24 19:52:00",//查询范围结束时间 10000,// 结果数据总量，默认1000行结果 True//排序，false正序排序，True为倒序排序 ] }} |
| 返回结果 | 类型 | Json 字符 |
| 格式举例 | {    "Data": "{\"results\":[{\"statement\_id\":0,\"series\":[{\"name\":\"TagRecord\_INT\",\"columns\":[\"time\",\"Name\",\"ParentPath\",\"QualityStamp\",\"Value\"],\"values\":[[\"2020-05-21T08:29:39.37Z\",\"IOTag1\",\"IO.Simulator1\",1,949],[\"2020-05-21T08:29:38.37Z\",\"IOTag1\",\"IO.Simulator1\",1,939],[\"2020-05-21T08:29:37.37Z\",\"IOTag1\",\"IO.Simulator1\",1,929],[\"2020-05-21T08:29:36.37Z\",\"IOTag1\",\"IO.Simulator1\",1,919],[\"2020-05-21T08:29:35.37Z\",\"IOTag1\",\"IO.Simulator1\",1,910],[\"2020-05-21T08:29:34.37Z\",\"IOTag1\",\"IO.Simulator1\",1,899],[\"2020-05-21T08:29:33.37Z\",\"IOTag1\",\"IO.Simulator1\",1,889],\"IOTag1\",\"IO.Simulator1\",99,0]]}]},{\"statement\_id\":1},{\"statement\_id\":2},{\"statement\_id\":3}]}\n",    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "af0c0b60-cb19-460c-bfa6-351995cacdb1",    "Error": null,    "IsSuccess": true} |
| 性能要求 | 根据不同的数据量返回的时间不同 |

### 2. 以DataTable的形式获取历史数据与内存数据

|  |  |
| --- | --- |
| 接口名称 | QueryToDT |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 第三方软件调用该接口，根据设备变量的路径集合查询历史数据库，以DataTable的形式返回历史数据和内存数据 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "QueryToDT",//查询方法名 "Paras": [["IO.Simulator1.IOTag1"],//设备变量路径集合 "2020-4-23 10:47:00",//查询范围开始时间 "2020-4-24 19:52:00",//查询范围结束时间 10000,// 结果数据总量，默认1000行结果 True//排序，false正序排序，True为倒序排序 ] }} |
| 返回结果 | 类型 | Json 字符 |
| 格式举例 | {    "Data": {        "TagRecord\_INT": [            {                "time": "2020-05-21 16:36:38:370",                "Name": "IOTag1",                "ParentPath": "IO.Simulator1",                "QualityStamp": 1,                "Value": 135            },            {                "time": "2020-05-21 16:36:37:370",                "Name": "IOTag1",                "ParentPath": "IO.Simulator1",                "QualityStamp": 1,                "Value": 125            },   "Timestamp": "0001-01-01T00:00:00",    "RequestId": "508956d7-01ee-4592-85f4-2a03143275a2",    "Error": null,    "IsSuccess": true} |
| 性能要求 | 根据不同的数据量返回的时间不同 |

### 3．以Serie的形式获取历史数据与内存数据

|  |  |
| --- | --- |
| 接口名称 | QueryToSER |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 第三方软件调用该接口，根据设备变量的路径集合查询历史数据库，以Serie的形式返回历史数据和内存数据 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "QueryToSER",//查询方法名 "Paras": [["IO.Simulator1.IOTag1"],//设备变量路径集合 "2020-4-23 10:47:00",//查询范围开始时间 "2020-4-24 19:52:00",//查询范围结束时间 10000,// 结果数据总量，默认1000行结果 True//排序，false正序排序，True为倒序排序 ] }} |
| 返回结果 | 类型 | Json 字符 |
| 格式举例 | {    "Data": [        {            "Name": "TagRecord\_INT",            "Tags": {},            "Columns": [                "time",                "Name",                "ParentPath",                "QualityStamp",                "Value"            ],            "Values": [                [                    "2020-05-21 16:52:38:370",                    "IOTag1",                    "IO.Simulator1",                    1,                    723                ],                [                    "2020-05-21 16:52:37:370",                    "IOTag1",                    "IO.Simulator1",                    1,                    713                ],],            "KeyIndex": 0        }    ],    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "5b3edb36-6315-407e-95f9-89aadd73031c",    "Error": null,    "IsSuccess": true} |
| 性能要求 | 根据不同的数据量返回的时间不同 |

### 4.获取离线数据

|  |  |
| --- | --- |
| 接口名称 | QueryCachefile |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 第三方软件调用该接口，根据设备变量的路径集合查询内存数据，以Serie的形式返回离线数据 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "QueryCachefile",//查询方法名 "Paras": [["IO.Simulator1.IOTag1"],//设备变量路径集合 "2020-4-23 10:47:00",//查询范围开始时间 "2020-4-24 19:52:00",//查询范围结束时间 10000,// 结果数据总量，默认1000行结果 True//排序，false正序排序，True为倒序排序 ] }} |
| 返回结果 | 类型 | Json 字符 |
| 格式举例 | {    "Data": [        {            "Name": "TagRecord\_INT",            "Tags": {                "Name": "IOTag1",                "ParentPath": "IO.Simulator1"            },            "Columns": [                "time",                "Name",                "ParentPath",                "QualityStamp",                "Value"            ],            "Values": [                [                    "2020-05-21 17:00:37:370",                    "IOTag1",                    "IO.Simulator1",                    1,                    512                ],                [                    "2020-05-21 17:00:36:370",                    "IOTag1",                    "IO.Simulator1",                    1,                    502                ]            ],            "KeyIndex": 0        }    ],    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "8b960478-0fa3-4712-a59e-2227316c1a1b",    "Error": null,    "IsSuccess": true} |

### 5. 聚合查询

|  |  |
| --- | --- |
| 接口名称 | QueryByInterval |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 第三方软件调用该接口，根据设备变量的路径集合以设定的间隔通过数据聚合函数获取聚合结果 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "QueryByInterval",//查询方法名 "Paras": [["IO.Simulator1.IOTag1"],//设备变量路径集合 "2020-5-21 09:00:00",//开始时间 "2020-5-21 18:37:00",//结束时间 15,//聚合参数类型 1,//间隔 1,//时间精度 10000,//最大返回数 False//排序方式 ] }} |
| 返回结果 | 类型 | Json 字符 |
| 格式举例 | {    "Data": [        {            "Name": "TagRecord\_INT",            "Tags": {                "Name": "IOTag1",                "ParentPath": "IO.Simulator1"            },            "Columns": [                "time",                "last",                "first",                "max",                "min",                "mean"            ],            "Values": [                [                    "2020-05-21 12:00:00:000",                    478,                    0,                    1000,                    0,                    474.77668539325845                ]      "KeyIndex": 0        }    ],    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "c54dbf25-f7cb-4739-a211-a80365193a6e",    "Error": null,    "IsSuccess": true} |

## 三.设备报警及处理接口

### 1.添加报警记录

|  |  |
| --- | --- |
| 接口名称 | AddAlarm |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 产生一条报警记录 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "AddAlarm",//方法名 "Paras": [{"AlarmType":"1",//报警类型 "AlarmLevel":"Info",//报警等级 "AlarmValue":0,//报警值 "LimitValue":0,//限值 "AlarmSource":"DESKTOP-POAVADV",//报警员 "AlarmContent":"",//报警内容 "LinkAlarmPath":"IO.Simulator1.IOTag1"//关联变量路径 } ] }} |
| 返回结果 | 类型 | Json 字符 |
| 格式举例 | {    "Data": {        "AlarmRuleType": 0,        "AlarmID": 1000000,        "LinkAlarmPath": "IO.Simulator1.IOTag1",        "AlarmType": "1",        "AlarmLevel": 1,        "AlarmValue": 0,        "LimitValue": 0,        "RecoveryValue": null,        "AlarmStatus": 1,        "AlarmSource": "DESKTOP-POAVADV",        "AlarmContent": "",        "AlarmReason": null,        "TriggerTime": "2020-05-22T09:29:56.5456802+08:00",        "AckedTime": "0001-01-01T00:00:00",        "RecoveryTime": "0001-01-01T00:00:00",        "AckReason": null,        "AckSource": null,        "ImproveStrategy": null,        "Remark": null,        "Description": null,        "TimeStamp": "2020-05-22T09:29:56.5456802+08:00"    },    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "7c14f627-628c-43ef-a09d-179abc1ec8f1",    "Error": null,    "IsSuccess": true} |

### 2.恢复报警

|  |  |
| --- | --- |
| 接口名称 | RecoveryAlarm |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 通过指定属性恢复报警 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "RecoveryAlarm",//方法名 "Paras": [{"AlarmType":"1",//报警类型 "AlarmSource":"DESKTOP-POAVADV",//报警源 "LinkAlarmPath":"IO.Simulator1.IOTag1"//关联变量路径 },5//恢复值 ] }} |
| 返回结果 | 类型 | 布尔值 |
| 格式举例 | {    "Data": true,    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "6447a683-d50f-47cc-aad3-6156842d1417",    "Error": null,    "IsSuccess": true} |

### 3. 应答报警

|  |  |
| --- | --- |
| 接口名称 | AckedAlarm |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 通过指定属性恢复报警 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "AckedAlarm", "Paras": [{"AlarmType":"1",//报警类型 "AlarmSource":"DESKTOP-POAVADV",//报警源 "LinkAlarmPath":"IO.Simulator1.IOTag1"//关联变量路径 } ] }} |
| 返回结果 | 类型 | 布尔值 |
| 格式举例 | {    "Data": true,    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "99a42539-b880-4e3f-a91f-ce26ceb2e9e6",    "Error": null,    "IsSuccess": true} |

### 4. 更新报警信息

|  |  |
| --- | --- |
| 接口名称 | UpdateAckReason |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 更新报警信息 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "UpdateAckReason", "Paras": [{"AlarmType":"1",//报警类型 "AlarmSource":"DESKTOP-POAVADV",//报警源 "LinkAlarmPath":"IO.Simulator1.IOTag1"//关联变量路径 },"液位恢复" ] }} |
| 返回结果 | 类型 | Json字符 |
| 格式举例 | {    "Data": true,    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "afa6e8e0-5f8b-44b7-b54d-e907aa4ee311",    "Error": null,    "IsSuccess": true} |

### 5. 订阅报警

|  |  |
| --- | --- |
| 接口名称 | SubscribeAlarm |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 订阅SMDC报警信息 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "SubscribeAlarm", "Paras": [ ] }} |
| 返回结果 | 类型 | Json字符 |
| 格式举例 | {    "Data": 1,    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "3f9335c1-0a6f-4fcb-8dd8-8fbe8f3940c7",    "Error": null,    "IsSuccess": true} |
| 性能要求 | 响应结果3秒内返回 |

### 6. 取消报警订阅

|  |  |
| --- | --- |
| 接口名称 | RemoveSubcribe |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 通过订阅的ID取消报警订阅 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "RemoveSubcribe", "Paras": [1//取消报警ID ] }} |
| 返回结果 | 类型 | Json字符 |
| 格式举例 | {    "Data": true,    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "a5370163-6fc5-4c7e-9905-cf3fde101f4d",    "Error": null,    "IsSuccess": true} |

### 7. 获取实时报警

|  |  |
| --- | --- |
| 接口名称 | GetAllAlarmRecords |
| 请求方式 | Post |
| 请求地址 | http://{host}/dataservice/webapi/rpcinvoke |
| 接口说明 | 通过订阅的ID获取所有的实时报警记录 |
| 接口提供者 | SMDC服务器 |
| 接口使用者 | 第三方软件 |
| 输入参数 | 名称 | 类型 | 说明 | 参数要求 |
|  | Json字符 | 设备变量的路径 | 必输 |
| 格式举例 | { "RpcPara": { "MethodName": "GetAllAlarmRecords", "Paras": [2//获取报警的订阅ID ] }} |
| 返回结果 | 类型 | Json字符 |
| 格式举例 | {    "Data": [        {            "AlarmRuleType": 0,            "AlarmID": 1000005,            "LinkAlarmPath": "IO.Simulator1.IOTag1",            "AlarmType": "1",            "AlarmLevel": 1,            "AlarmValue": 0,            "LimitValue": 0,            "RecoveryValue": null,            "AlarmStatus": 1,            "AlarmSource": "DESKTOP-POAVADV",            "AlarmContent": "",            "AlarmReason": null,            "TriggerTime": "2020-05-22T10:25:05.9990596+08:00",            "AckedTime": "2020-05-22T10:25:45.2888402+08:00",            "RecoveryTime": "0001-01-01T00:00:00",            "AckReason": "液位恢复",            "AckSource": null,            "ImproveStrategy": null,            "Remark": null,            "Description": null,            "TimeStamp": "2020-05-22T10:25:05.9990596+08:00"        }    ],    "Timestamp": "0001-01-01T00:00:00",    "RequestId": "e7062d96-ce73-452b-a8b3-c65ab5192fef",    "Error": null,    "IsSuccess": true} |