



**SmartGen**  
ideas for power

**CMM365-4G**  
云监控通信模块  
通信协议

SmartGen

郑州众智科技股份有限公司  
SMARTGEN (ZHENGZHOU) TECHNOLOGY CO., LTD.



## 目 次

1 引言 .....	6
2 JSON 基本规则 .....	6
3 数据帧格式 .....	6
4 云猫注册 .....	6
4.1 简介 .....	6
4.1.1 说明 .....	6
4.1.2 接口方向 .....	6
4.1.3 通讯协议 .....	6
4.1.4 接口定义 .....	7
4.2 云猫主动发动长连接 .....	8
4.2.1 简介 .....	8
4.2.2 接口方向 .....	8
4.2.2 通讯协议 .....	8
4.2.3 接口定义 .....	9
4.3 云猫发送心跳数据协议 .....	9
4.3.1 简介 .....	9
4.3.2 接口方向 .....	9
4.3.3 通信协议 .....	9
4.3.4 接口定义 .....	9
4.4 云猫实时上传监控数据（全量） .....	10
4.4.1 简介 .....	10
4.4.2 接口方向 .....	10
4.4.3 通信协议 .....	10
4.4.4 接口定义 .....	10
4.5 云猫实时上传监控数据（变化再传） .....	11
4.5.1 简介 .....	11
4.5.2 接口方向 .....	11
4.5.3 通信协议 .....	11
4.5.4 接口定义 .....	11
4.6 写配置项数据 .....	12
4.6.1 简介 .....	12
4.6.2 接口方向 .....	12
4.6.3 通信协议 .....	12



---

4.6.4 接口定义.....	13
4.7 授时接口.....	14
4.7.1 简介.....	14
4.7.2 接口方向.....	14
4.7.3 通信协议.....	14
4.7.4 接口定义.....	14
4.8 重置接口.....	15
4.8.1 简介.....	15
4.8.2 接口方向.....	15
4.8.3 通信协议.....	15
4.8.4 接口定义.....	15
4.9 数据协议.....	16

SmartGen

## 前 言



是众智的中文商标

**SmartGen**是众智的英文商标

**SmartGen** – Smart 的意思是灵巧的、智能的、聪明的，Gen 是 generator(发电机组)的缩写，两个单词合起来的意思是让发电机组变得更加智能、更加人性化、更好的为人类服务!

不经过本公司的允许，本文档的任何部分不能被复制（包括图片及图标）。

本公司保留更改本文档内容的权利，而不通知用户。

公司地址：中国·河南省郑州高新技术开发区金梭路 28 号

电话：+86-371-67988888/67981888/67992951

+86-371-67981000（外贸）

传真：+86-371-67992952

网址：[www.smartgen.com.cn/](http://www.smartgen.com.cn/)

[www.smartgen.cn/](http://www.smartgen.cn/)

邮箱：[sales@smartgen.cn](mailto:sales@smartgen.cn)

表1 版本发展历史

日期	版本	内容
2022-01-14	V1.0	开始发布。

本文档适用于

CMM366A-3G/CMM366A-ET/CMM366A-WIFI/CMM365A-2G/CMM366A-4G/CMM366B-4G/CMM365-4G 监控通讯模块。

本文档所用符号的说明：

表2 符号说明

符号	说明
 注意	该图标提示或提醒操作员正确操作。
 小心	该图标表示错误的操作有可能损坏设备。
 警告	该图标表示错误的操作有可能会造成死亡、严重的人身伤害和重大的财产损失。

SmartGen

## 1 引言

本通讯协议详细描述了本模块GPRS和云服务器通讯过程协议，通过本协议可以实现发电机组（带有串行通信接口）连入因特网。模块登录云服务器后，云服务器将传送相应的发电机组控制器通信协议到模块，云监控模块可通过RS485接口，USB接口，LINK接口，CAN接口或者RS232接口获取发电机组的数据信息，通过4G无线网络将获取的数据信息传送到相应的云服务器，用户可通过手机APP（IOS或安卓系统）、电脑等终端设备实时监控发电机组的运行状态和查询发电机组的运行记录。

## 2 JSON 基本规则

- JSON(JavaScript Object Notation): JSON 是一种轻量级的数据交换格式，基于 ECMAScript 的一个子集，采用完全独立于编程语言的文本格式来存储和表示数据。
- JSON 数据书写格式：名称/值对；
- JSON 可以是数字、字符串、逻辑值、数组、对象、null。

## 3 数据帧格式

通讯双方有发起方：Server；接收方：Device，通讯规则用 json 格式。

## 4 云猫注册

### 4.1 简介

#### 4.1.1 说明

云监控模块向服务器发送注册信息，服务器根据 Hostid 验证设备是否注册过，如果已注册则执行更新操作，未注册则执行保存操作，云监控模块并将实时数据上传服务器 IP 和端口、历史数据上传服务器 IP 和端口及该设备型号对应的参数返回至云猫。

#### 4.1.2 接口方向

发起方：Device；接收方：Server。

#### 4.1.3 通讯协议

地址端口：ServerIp

协议：Socket

方法：短连接

传输格式：JSON

#### 4.1.4 接口定义

表3 消息请求

序号	名称	必填	类型	最大长度	参数说明
1	method	Y	String	20	login
2	params	Y	String		参数 JSON
2.1	Hostid	Y	String	24	云猫 ID(24 位)
2.2	hostname	Y	String	20	云猫名称(UTF-8)
2.3	password	Y	String	20	云猫登录密码
2.4	longitude	N	String	20	云猫经度
2.5	latitude	N	String	20	云猫纬度
2.6	Gpsen	N	Num	1	0: 百度坐标; 1: gps 坐标。
2.7	altitude	N	String	20	云猫海拔
2.8	moduletype	Y	String	30	模块型号 (用于区别不同控制器) 详见: 机组控制器型号列表
2.9	modulePort	Y	Num	2	0: 不使能; 1: LINK; 2: RS483; 3: RS232; 4: USB
2.10	moduleBaud	Y	Num	0	0: 9600; 1: 19200; 2: 115200 bit/s

消息请求示例:

```
{
  "method": "login",
  "params": {
    "hosted": "AAAAA",
    "hostname": "Smatgen",
    "password": "980318",
    "longitude": "54",
    "latitude": "0",
    "gsen": 0,
    "moduletype": "HGM6120"
  }
}
```

表4 消息响应

序号	名称	类型	参数说明	
1.1	Method	String	login	
1.2	Retcode	String	参考返回状态码	
1.3	Result			
1.3.1	register	Int	注册状态 1 成功 0 失败	
1.3.2	historic	String	历史数据上传地址端口	
1.3.3	liveData	String	实时数据上传地址端口	
1.3.4	realtime	String	返回时间 UTC	
1.3.6	para_command	String	监控项目命令, 多个命令使用分号分开(;) )	控制器型号, 对应不同的代码
1.3.7	con_command	String	读取配置项命令, 多个命令	

序号	名称	类型	参数说明	
			使用分号分开(;)	参照 1.3.6
1.3.8	online_rate	Int	客户端在线发送频率	讨论结果：例如传 3000
1.3.9	offline_rate	Int	客户端未在线发送频率	讨论结果：和 online_rate 一样
1.3.11	moduletype	String	模块型号	传控制器的型号
1.3.12	data_mode	Int	数据上传模式	0：省流量模式；（详见 3.5） 1：全传模式（详见：3.4）
1.3.13	modulePort	Int	Num	0：不使能； 1:LINK;2:RS483;3:RS232;4:USB
1.3.14	moduleBaud	Int	Num	0:9600;1:19200;2:115200 bit/s

消息响应示例：

```
{
  "method": "login",
  "result": {
    "register": 1,
    "historic": "192.168.0.194:81",
    "liveData": "192.168.0.194:81",
    "realTime": "UTC",

    "para_command": "01030000005045F6; 01030000005045F6",
    "con_command": "01030000005045F6; 01030000005045F6",
    "online_rate": 3000,
    "offline_rate": 3000,
    "modulePort": 2,
    "moduleBaud": 0,
  }
  "retcode": "000000"
}
```

## 4.2 云猫主动发动长连接

### 4.2.1 简介

云猫在注册完毕之后，主动向服务器发送长连接请求，该连接创建完成之后，服务端返回连接成功的状态。

### 4.2.2 接口方向

发起方：Device；接收方：Server。

### 4.2.2 通讯协议

地址端口：ServerIp  
 协议：Socket  
 方法：长连接  
 传输格式：JSON

### 4.2.3 接口定义

表5 消息请求

序号	名称	必填	类型	最大长度	参数说明
1	method	Y	String	20	LongCon
2	hostid	Y	String	24	云猫 ID (24 位)

消息请求示例:

```
{"method": "LongCon", "hostid": "AAAAAAAAAA"}
```

表6 消息响应

序号	名称	类型	参数说明
1	retcode	Y	String
2	message	String	消息
3	method	String	LongCon

消息响应示例:

```
{"method": "LongCon", "message": "OK", "retcode": "000000"}
```

## 4.3 云猫发送心跳数据协议

### 4.3.1 简介

云猫在注册完毕之后, 定时向服务器发送心跳数据, 检查网络的在线状态。

### 4.3.2 接口方向

发起方: Device; 接收方: Server。

### 4.3.3 通信协议

地址端口: ServerIp

协议: Socket

方法: 长连接

传输格式: JSON

### 4.3.4 接口定义

表7 消息请求

序号	名称	必填	类型	最大长度	参数说明
1	method	Y	String	20	HB
2	params	Y	String		Null
3	hostid	Y	String	24	云猫 ID (24 位)

消息请求示例:

```
{
  "method": "HB",
  "hostid": "AAAAAAAAAA",
  "params": ""
}
```

表8 消息响应

序号	名称	类型	参数说明
1	retcode	String	状态
2	message	String	消息
3	method	String	HB

消息响应示例:

```
{"method": "HB", "message": "OK", "retcode": "000000"}
```

#### 4.4 云猫实时上传监控数据（全量）

##### 4.4.1 简介

云猫根据当前客户端是否在线设置不同的数据上传频率。请求结果同时保存到内存数据库和历史数据库。

内存数据存储格式: monitor\_hostid: {id, ymid, hostid, data, mask, status, itime, device No}

当客户端请求实时监控数据时, 服务器直接从内存数据库中进行读取。

历史数据的保存, 需要根据上一次保存历史数据的时间和当前时间的差与历史数据保存频率进行对比, 当时间差大于历史数据的保存时间时才进行保存操作。并将当前时间记录为最近一次历史数据的保存时间。

##### 4.4.2 接口方向

发起方: Device; 接收方: Server。

##### 4.4.3 通信协议

地址端口: ServerIp

协议: Socket

方法: 长链接

传输格式: JSON, 命令内容为 ModBus

##### 4.4.4 接口定义

表9 消息响应

序号	名称	类型	参数说明
1	method	String	接口类型, reqdata 请求数据
2	params	Object	ModBus 数据的内容, 多条数据使用分号(;)分开
3	hostid	String	猫的 ID

消息响应示例:

```
{
  "method": "reqdata",
  "params": "",
  "hostid": "AAAAAAA"
}
```

表10 消息请求

序号	名称	类型	参数说明
1	retcode	String	状态
2	message	String	消息
3	method	String	接口类型, reqdata 请求数据

消息请求示例:

```
{"method": "reqdata", "message": "OK", "retcode": "000000"}
```

#### 4.5 云猫实时上传监控数据 (变化再传)

##### 4.5.1 简介

云猫根据当前客户端设置的数据上传频率, 将变化的数据上传到云服务器。

##### 4.5.2 接口方向

发起方: Device; 接收方: Server。

##### 4.5.3 通信协议

地址端口: ServerIp

协议: Socket

方法: 长连接

传输格式: JSON, 命令内容为 ModBus

##### 4.5.4 接口定义

表11 消息响应

序号	名称	类型	参数说明
1	method	String	接口类型, reqdatachange 请求数据
2	params	Object	只上传变化的数据参数
3	hostid	String	猫的 ID
4	AccessAlarm	Int	门禁报警输入 0: 无效; 1: 有效
5	FireAlarm	Int	消防报警输入 0: 无效; 1: 有效
6	AlarmInput	Int	报警输入 0: 无效; 1: 有效
7	InhbitRemote	Int	远程控制禁止输入 0: 无效; 1: 有效

消息响应示例:

```
{
  "method": "reqdatachange",
  "params": {
    "03": [
      {
```



```

        "0000": "0805"
    }
},
"01": [
    {
        "0000": "0805"
    }
]
},
"hostid": "AAAAAAA"
}

```

表12 消息请求

序号	名称	类型	参数说明
1	retcode	String	状态
2	message	String	消息
3	method	String	接口类型, reqdata 请求数据

消息请求示例:

```
{"method": "reqdata", "message": "OK", "retcode": "000000"}
```

## 4.6 写配置项数据

### 4.6.1 简介

云猫在接收到写配置项命令后，主动向控制器发送写配置命令，当云猫接收到控制器的返回结果时，将命令和执行状态进行拼接（用逗号分开，比如：0000000000,1）。

### 4.6.2 接口方向

发起方： Server；接收方： Device。

### 4.6.3 通信协议

地址端口： ServerIp

协议： Socket

方法： 长链接

传输格式： JSON，命令内容为 ModBus

#### 4.6.4 接口定义

表13 消息请求

序号	名称	类型	参数说明	备注
1	result	String	ModBus 协议的内容	根据控制器系列定义
2	datatype	String	ModBus 数据格式	根据控制器系列定义
3	uid	String	请求人的用户 id	任意传
4	method	String	请求方法 writeConfig	
5	modulePort	Num	0: 不使能; 1: LINK; 2: RS483; 3: RS232; 4: USB	
6	moduleBaud	Num	0:9600; 1: 19200; 2: 115200 bit/s	

写配置消息请求示例:

```
{
  "result": "000000000000",
  "datatype": 1,
  "uid": "1",
  "method": "writeConfig"
}
```

写云猫端口号或波特率消息请求示例:

```
{
  "modulePort": 1,
  "moduleBaud": 0,
  "datatype": 1,
  "uid": "1",
  "method": "writeConfig"
}
```

表14 消息响应

序号	名称	类型	参数说明
1	method	String	接口类型, writeConfig 请求数据
2	params	Object	命令, 状态 比如: (0000000000,1)1 表示成功 0 表示失败
3	hostid	String	猫的 ID
4	uid	String	请求用户的 id
5	modulePort	Num	0: 不使能; 1: LINK; 2: RS483; 3: RS232; 4: USB
6	moduleBaud	Num	0: 9600; 1: 19200; 2: 115200 bit/s

写配置消息响应示例:

```
{
  "method": "writeConfig",
  "params": "01010101,1",
  "hostid": "AAAAA",
  "uid": "1"
}
```



写云猫端口号或波特率消息响应示例：

```
{
  "method": "writeConfig",
  "modulePort": 1,
  "moduleBaud": 0,
  "hostid": "AAAAAAA",
  "uid": "1"
}
```

## 4.7 授时接口

### 4.7.1 简介

云猫定时向服务器请求网络授时来保证云猫与服务器时钟同步。

### 4.7.2 接口方向

发起方：Device；接收方：Server。

### 4.7.3 通信协议

地址端口：ServerIp

协议：Socket

方法：长链接

传输格式：JSON

### 4.7.4 接口定义

表15 消息响应

序号	名称	类型	参数说明
1	method	String	接口类型，timing 请求数据
2	hostid	String	猫 ID

消息响应示例：

```
{
  "method": "timing",
  "hostid": "AAAAAAA",
}
```

表16 消息请求

序号	名称	类型	参数说明
1	message	String	消息提示内容
2	retcode	String	消息状态
3	method	String	请求方法 timing
4	result	String	当前时间 UTC long

消息请求示例：

```
{
  "message": "ok",
  "retcode": "000000",
}
```



```

    "method": "timing",
    "result": "1497811080077"
  }

```

## 4.8 重置接口

### 4.8.1 简介

配置云猫；  
服务器发送重置命令后，云猫自动走注册功能。

### 4.8.2 接口方向

发起方：Server；接收方：Device。

### 4.8.3 通信协议

地址端口：ServerIp

协议：Socket

方法：长链接

传输格式：JSON

### 4.8.4 接口定义

表17 消息请求

序号	名称	类型	参数说明
1	message	String	消息提示内容
2	retcode	String	消息状态
3	method	String	请求方法 reset

消息请求示例：

```

{
  "message": "ok",
  "retcode": "000000",
  "method": "reset",
}

```

表18 消息响应

序号	名称	类型	参数说明
1	method	String	接口类型，reset 请求数据
2	hostid	String	猫的 ID

消息响应示例：

```

{
  "method": "reset",
  "hostid": "AAAAAAA",
}

```

#### 4.9 数据协议

表19 云猫向服务器发送数据格式 (Socket)

序号	名称	类型	参数说明
1	method	String	接口类型
2	params	Object	数据对象，可以是字符串，也可以是 Json 对象、也可以是 ModBus 的协议内容
3	hostid	String	猫的 ID
4	uid	String	用户的 ID

表20 服务器向云猫发送数据格式 (Socket)

序号	名称	类型	参数说明
1	result	Object	数据对象，可以是字符串，也可以是 Json 对象、也可以是 ModBus 的协议内容
2	retcode	String	
3	uid	String	用户的 ID (6 位)
4	datatype	Int	result 数据类型，0 字符串数据 1 ModBus 数据 2 表示 Json 对象
5	message	String	可能为空
6	packetnum	Int	参数包号