

SGA-100X多参数大气模块通讯协议

Ver: 181101

一、基本协议

- 1.1、支持485 半双工通信方式通信。
- 1.2、通信格式为MODBUS-RTU 方式，8 位数据位，1 位停止位，无校验。波特率：9600BPS。
- 1.3、支持03 号和06 号两条MODBUS 命令。
- 1.4、通信时为从机方式。主机提出命令请求，控制器响应；接收数据后做数据分析，如果数据满足通信规约，从机做出响应。（采样周期>1000MS，以免过于频繁响应中断影响模块采样浓度周期）从接收完命令帧到开始应答的时间小于200MS。若从机检测数据错，或不及时响应主机，主机做超时处理。
- 1.5、数据帧格式：

从机地址	命令字	信息字	校验码
------	-----	-----	-----

从机地址：（1 个字节）：从机设备号，主机利用从机地址来识别进行通讯的从机设备。表明由用户设置地址的从机，将接收由主机发送来的信息。每个从机都必须有唯一的地址码，并且只有符合地址码的从机才能响应回送。FE为广播地址，所有模块均会响应，使用广播地址时485总线上只能连接1套模块。

命令字：（1 个字节）：主机发送的功能码，告诉从机执行什么任务。

信息字：（N 个字节）：包括进行两机通讯中各种数据信息，数据长度，读写的数据等。

校验码：（2 个字节）：用于检测数据通信错误，采用循环冗余码CRC16，详见附件2。

1.6、通信命令, 仅支持03和06指令：

1) 读输入/保持寄存器： 03，命令：

从机地址	03	寄存器地址 高位	寄存器地址 低位	寄存器数量 高位	寄存器数量 低位	校验码	校验码
		ADRH	ADRL	N_H	N_L	CRC_L	CRC_H

正确应答：

从机地址	03	数据字节数	数据	校验码	校验码
		N_L*2	N_L*2 字节数据	CRC_L	CRC_H

2) 单寄存器写：06命令。

从机地址	06	寄存器地址 高位	寄存器地址 低位	寄存器值 高位	寄存器值 低位	校验码	校验码
		ADRH	ADRL	N_H	N_L	CRC_L	CRC_H

正确应答：（原数据返回）

从机地址	06	寄存器地址 高位	寄存器地址 低位	寄存器值 高位	寄存器值 低位	校验码	校验码
		ADRH	ADRL	N_H	N_L	CRC_L	CRC_H

二、数据地址表及功能说明： (1个寄存器2字节)

功能码	变量功能		指令起始地址	数据长度 (寄存器数量)	变量功能说明
03 功能码	系统参数	模块地址	0X00F0	1	0X0000-0X0093
		传感器数量	0X00F1	1	0X0000-0X000F
		上传模式	0X00F4	1	01: 主动上传模式 00: 主从模式
	传感器多 参数值 (可一次 性读取这 (N+4)*5 个寄存 器)	1#传感器参数	0X0500	5	5个寄存器依次 为: 状态、浓度、小数 点位数、气体名 称、单位 数据格式见: 传感 器参数说明
		2#传感器参数	0X0500+5	5	
			
		N#传感器参数	0X0500+(N-1) *5	5	
		温度值	0X0500+(N+0) *5	5	
		湿度值	0X0500+(N+1) *5	5	
		PM2.5测量值	0X0500+(N+2) *5	5	
		PM10测量值	0X0500+(N+3) *5	5	
	传感器 测量值 (可一次 性读取这 N+4个寄 存器)	1#传感器测量值	0X0600	1	0-50000 (0X1388)
		2#传感器测量值	0X0600+1	1	0-50000 (0X1388)
		1	
		N#传感器测量值	0X0600+(N-1)	1	0-50000 (0X1388)
		温度测量值	0X0600+N+0	1	有符号数, 16BITS 最高位为数符 (1 负数, 0整数) 其他位是绝对值 (默认1位小数)
		湿度测量值	0X0600+N+1	1	0-50000 (0X1388)
		PM2.5测量值	0X0600+N+2	1	0-50000 (0X1388)
		PM10测量值	0X0600+N+3	1	0-50000 (0X1388)

06 功能 码	系统参数	模块地址	0X30F0	1	0X0000-0X0093
		传感器数量	0X30F1	1	0X0000-0X000F
		上传模式	0X30F4	1	写入01: 主动上传 写入00: 主从模式
	1号传感器 参数设置	高限报警值	0X3100	1	0-50000 (0X1388)
		低限报警值	0X3101	1	0-50000 (0X1388)
		调零基准值	0X3102	1	第1字节: 固定01; 第2字节: 基准值 (范围00-255)
		标定值	0X3103	1	0-50000 (0X1388)
		调零	0X3104	1	写入: 0X00AA
		恢复出厂值	0X3105	1	写入: 0X00AA
	2号传感 参数设置		0X3200-0x3205	6	6个参数均与1号 参数一致 N本系统最大为6 个传感器
		
	N号传感 器参数设 置		0X3N00-0x3N05	6	

注意：传感器数量N指的是智能传感器数量，不包括PM2.5和温湿度传感器。用户需根据系统所带智能传感器数量适当修改寄存器的首地址。

03读取命令通讯例举：

2. 1、读取1-10#传感器多参数值（开通6个智能传感器）

发送：01 03 05 00 00 32 C4 D3

返回：01 03 64 00 00 00 67 00 02 00 02 00 02 00 01 00 D1 00 03 00 0A 00
02 00 00 00 4C 00 03 00 16 00 02 00 02 02 0B 00 03 00 08 00 02 00 00 00
21 00 03 00 27 00 02 00 04 00 00 00 00 00 00 00 04 00 00 01 2C 00 01 00
83 00 05 00 00 01 FE 00 01 00 84 00 06 00 00 00 AD 00 00 00 81 00 07 00
00 00 B9 00 00 00 82 00 07 B7 87

0500-0504共5个寄存器：1#智能传感器；正常1.03PPM一氧化碳；

0505-0509共5个寄存器：2#智能传感器；低限报警0.209PPM二氧化硫；

050A-050E共5个寄存器：3#智能传感器；正常0.076PPM二氧化氮；

050F-0513共5个寄存器：4#智能传感器；高限报警0.523PPM臭氧；

0514-0518共5个寄存器：5#智能传感器；正常0.033PPM VOC；

0519-051D共5个寄存器：6#智能传感器；通讯故障；

051E-0522共5个寄存器为：温度；正常30.0℃；

0523-0527共5个寄存器为：湿度；正常51.0%RH；

0528-052C共5个寄存器为：PM2.5；正常173 μg/m3；

052D-0531共5个寄存器为：PM10；正常185 μg/m3；

00 00 00 67 00 02 00 02 00 02；

0000状态：正常；0067浓度：103；0002小数位2位；0002名称：一氧化碳；0002

单位：PPM。此数据代表的是正常工作1.03PPM的一氧化碳。

后面的每组数据都按这个解析。

当状态值为0004通讯故障时，此传感器其它数据均可不处理。

2.2、读取1-9#传感器多参数值（开通5个智能传感器）

发送：01 03 05 00 00 2D 85 1B

返回：01 03 5A 00 01 00 D1 00 03 00 0A 00 02 00 00 00 4C 00 03 00 16 00 02 00 02 02 0B 00 03 00 08 00 02 00 00 00 21 00 03 00 27 00 02 00 04 00 00 00 00 00 00 04 00 00 01 2C 00 01 00 83 00 05 00 00 01 FE 00 01 00 84 00 06 00 00 00 AD 00 00 00 81 00 07 00 00 00 B9 00 00 00 82 00 07 82 34

2.3、读取9-10#传感器多参数值（开通6个智能传感器，只读PM2.5、PM10）

发送：01 03 05 28 00 0A 45 09

返回：01 03 14 00 00 00 AD 00 00 00 81 00 07 00 00 00 B9 00 00 00 82 00 07 61 C3

2.4、读取10#传感器多参数值（开通6个智能传感器，只读PM10）

发送：01 03 05 2D 00 05 15 0C

返回：01 03 0A 00 00 00 B9 00 00 00 82 00 07 ED 97

2.5、读取1-10#传感器测量值（开通6个智能传感器）

发送：01 03 06 00 00 0A C5 45

返回：01 03 14 00 D7 00 00 00 00 00 00 00 00 01 2C 01 FD 00 87 00 C3 00 CD 79 80

2.6、读取1-9#传感器测量值（开通5个智能传感器）

发送：01 03 06 00 00 09 85 44

返回：01 03 12 00 D7 00 00 00 00 00 00 00 00 01 2E 01 FB 00 F4 01 26 37 39

2.7、读取6-7#传感器测量值（开通5个智能传感器，只读温、湿度）

发送：01 03 06 05 00 02 D4 82

返回：01 03 04 01 2E 01 FB DB D5

2.8、读取多参数模块地址（FE广播地址）

发送：FE 03 00 F0 00 01 90 36

返回：01 03 02 00 01 79 84

2.9、读取多参数模块传感器数量

发送：01 03 00 F1 00 01 D5 F9

返回：01 03 02 00 06 38 46

2.10、读取多参数模块上传模式

发送：01 03 00 F4 00 01 C5 F8

返回：01 03 02 00 00 B8 44

06写寄存器命令通讯例举：

2.11、修改多参数模块地址1#（FE广播地址）

发送：FE 06 30 F0 00 01 53 36

返回：FE 06 30 F0 00 01 53 36

2.12、修改多参数模块传感器数量为5

发送：01 06 30 F1 00 05 17 3A

返回：01 06 30 F1 00 05 17 3A

2.13、修改1#智能传感器高报值400

发送：01 06 31 00 01 90 86 CA

返回：01 06 31 00 01 90 86 CA

2.14、修改1#智能传感器低报值100

- 发送: 01 06 31 01 00 64 D7 1D
 返回: 01 06 31 01 00 64 D7 1D
- 2.15、修改1#智能传感器调零基准值50（调零后，显示值才能更改为设置后的值）
 发送: 01 06 32 01 01 32 56 F7
 返回: 01 06 32 01 01 32 56 F7
- 2.16、对1#智能传感器进行校准200
 发送: 01 06 31 03 00 C8 76 A0
 返回: 01 06 31 03 00 C8 76 A0
- 2.17、对1#智能传感器进行调零
 发送: 01 06 31 04 00 AA 46 88
 返回: 01 06 31 04 00 AA 46 88
- 2.18、对1#智能传感器恢复出厂值
 发送: 01 06 31 05 00 AA 17 48
 返回: 01 06 31 05 00 AA 17 48
- 2.19、修改多参数模块上传模式为主动上传（固定每5秒上传一组传感器测量值）
 发送: 01 06 30 F4 00 01 06 F8
 返回: 01 06 30 F4 00 01 06 F8
- 2.20、修改多参数模块上传模式为主从模式
 发送: 01 06 30 F4 00 00 C7 38
 返回: 01 06 30 F4 00 00 C7 38

三、传感器参数说明

- 3.1、传感器状态: ; 1 寄存器十六进制（2 字节）
 0X0000: 正常; 0X0001: 低限报警;
 0X0002: 高限报警: 2; 0X0003: 未定义;
 0X0004: 通讯故障: 4; 0X0005: 未定义;
 0X0006: 传感器内部故障;
- 3.2、测量值: 0-50000（0X0000-0XC350） ; 2 寄存器十六进制（4 字节）
- 3.3、小数点位置: 0-4 ; 1 寄存器十六进制（2 字节）
 0X0000: 整数; 0X0001: 1 位小数点;
 0X0002: 2 位小数点; 0X0003: 3 位小数点;
 0X0004: 4 位小数点;
- 3.4、物质名称: 0X0000-0X00FF ; 1 寄存器十六进制（2 字节）
 详见附件 1: 物质名称列表。
- 3.5、单位设置: ; 1 寄存器十六进制（2 字节）
 0X0000: %LEL; 0X0001: %VOL;
 0X0002: PPM; 0X0003: PPb;
 0X0004: 无单位(不显示) ; 0X0005: °C;
 0X0006: %RH; 0X0007: μg/m3;
 0X0008: mg/m3; 0X0009: MP;
 0X000A: L/Min; 0X000B: mL/Min;
 0X000C: L/h;

附件 1：物质名称列表

00 无	20 乙炔	40 四氯乙烯		80 PM1.0
01 可燃气体	21 乙烯	41 亚硫酸氯		81 PM2.5
02 一氧化碳	22 甲醛	42 乙酸乙烯酯		82 PM10
03 氧气	23 液化石油气	43 硫醇 TBM		83 温度
04 氢气	24 碳氢	44 TVOC		84 湿度
05 甲烷	25 苯	45 环己烷		85 压力
06 丙烷	26 过氧化氢	46 三氯乙烯		85 流量
07 二氧化碳	27 VOC	47 二甲苯		
08 臭氧	28 六氟化硫	48 氟利昂		
09 硫化氢	29 甲苯	49 一氯甲烷		
0A 二氧化硫	2A 联乙烯	4A 二氯甲烷		
0B 氨气	2B 氧硫化碳	4B 三氯甲烷		
0C 氯气	2C 联氨	4C 一甲胺		
0D 环氧乙烷	2D 硒化氢	4D 正戊烷		
0E 氯化氢	2E 苯乙烯	4E 正己烷		
0F 磷化氢	2F 异丁烯	4F 正庚烷		
10 溴化氢	30 亚甲基	50 异辛烷		
11 氰化氢	31 笑气	51 乙烷		
12 三氢化砷	32 天然气	52 石油醚		
13 氟化氢	33 光气	53 丁烷		
14 溴气	34 氯乙烯	54 乙醚		
15 一氧化氮	35 甲醇			
16 二氧化氮	36 乙醇			
17 氮氧化物	37 异丙醇			
18 二氧化氯	38 丙酮			
19 硅烷	39 乙醛			
1A 二硫化碳	3A 丙烯腈			
1B 氟气	3B 二甲基硫醚			
1C 乙硼烷	3C 环氧氯丙烷			
1D 锆烷	3D 乙酸乙酯			
1E 氮气	3E 甲基乙基酮			
1F 四氢噻吩	3F 甲硫醇			

```
#ifndef _MB_CRC_H
#define _MB_CRC_H

USHORT      usMBCRC16( UCHAR * pucFrame, USHORT usLen );

#endif
```

```
static const UCHAR aucCRCLo[] = {
    0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7,
    0x05, 0xC5, 0xC4, 0x04, 0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E,
    0x0A, 0xCA, 0xCB, 0x0B, 0xC9, 0x09, 0x08, 0xC8, 0xD8, 0x18, 0x19, 0xD9,
    0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE, 0xDF, 0x1F, 0xDD, 0x1D, 0x1C, 0xDC,
    0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2, 0x12, 0x13, 0xD3,
    0x11, 0xD1, 0xD0, 0x10, 0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32,
    0x36, 0xF6, 0xF7, 0x37, 0xF5, 0x35, 0x34, 0xF4, 0x3C, 0xFC, 0xFD, 0x3D,
    0xFF, 0x3F, 0x3E, 0xFE, 0xFA, 0x3A, 0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38,
    0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B, 0x2A, 0xEA, 0xEE, 0x2E, 0x2F, 0xEF,
    0x2D, 0xED, 0xEC, 0x2C, 0xE4, 0x24, 0x25, 0xE5, 0x27, 0xE7, 0xE6, 0x26,
    0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0, 0xA0, 0x60, 0x61, 0xA1,
```

```

0x63, 0xA3, 0xA2, 0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4,
0x6C, 0xAC, 0xAD, 0x6D, 0xAF, 0x6F, 0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB,
0x69, 0xA9, 0xA8, 0x68, 0x78, 0xB8, 0xB9, 0x79, 0xBB, 0x7B, 0x7A, 0xBA,
0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C, 0xB4, 0x74, 0x75, 0xB5,
0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0,
0x50, 0x90, 0x91, 0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97,
0x55, 0x95, 0x94, 0x54, 0x9C, 0x5C, 0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E,
0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59, 0x58, 0x98, 0x88, 0x48, 0x49, 0x89,
0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D, 0x4D, 0x4C, 0x8C,
0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83,
0x41, 0x81, 0x80, 0x40

```

```
};
```

```

usMBCRC16( UCHAR * pucFrame, USHORT usLen )
{
    UCHAR          ucCRCHi = 0xFF;
    UCHAR          ucCRCLo = 0xFF;
    int            iIndex;

    while( usLen-- )
    {
        iIndex = ucCRCLo ^ *( pucFrame++ );
        ucCRCLo = ( UCHAR )( ucCRCHi ^ aucCRCHi[iIndex] );
        ucCRCHi = aucCRCLo[iIndex];
    }
    return ( USHORT )( ucCRCHi << 8 | ucCRCLo );
}

```