Revision Log

By	Ver	Date	Remark
	V1.0	2016.9.7	Initial release
	V1.2.1	2018.8.20	
	V1.4.5	2019.3.15	
	V1.4.6	2019.3.27	CMD_READ_SENSOR_ID CMD_WRITE_SENSOR_ID added
	V1.4.7		
	V1.4.8	2019.07.06	*CMD_GET_MulCH_OFFSET = 0x2C,* *CMD_SET_MulCH_OFFSET = 0x2D,* *CMD_GET_PM25_OFFSET = 0x2E,* *CMD_SET_PM25_OFFSET = 0x2F,* added
	V1.4.9	2019.07.29	
	V1.5.0	2019.08.16	Add usr_path[128]
	V1.5.1	2019.08.20	eWH57_SENSOR, eWH55_SENSORCH1, eWH55_SENSORCH2, eWH55_SENSORCH3, eWH55_SENSORCH4, #define ITEM_LEAK_CH1 0x58//for Leak_ch1 #define ITEM_LEAK_CH2 0x59//for Leak_ch2 #define ITEM_LEAK_CH3 0x5A//for Leak_ch3 #define ITEM_LEAK_CH4 0x5B//for Leak_ch4 #define ITEM_THUNDERDISTANCE 0x60 added
	V1.5.2	2020.04.30	Add ITEM_RAINEVENT data output
	V1.5.3	2020.05.06	#define ITEM_TF_USR1 0x63//Temperature(℃) 3Byte #define ITEM_TF_USR2 0x64//Temperature(℃) 3Byte #define ITEM_TF_USR3 0x65//Temperature(℃) 3Byte #define ITEM_TF_USR4 0x66//Temperature(℃) 3Byte #define ITEM_TF_USR5 0x67//Temperature(℃) 3Byte #define ITEM_TF_USR6 0x68//Temperature(℃) 3Byte #define ITEM_TF_USR7 0x69//Temperature(℃) 3Byte #define ITEM_TF_USR8 0x6A//Temperature(℃) 3Byte #define ITEM_TF_BATT 0x6B//tf temperature batt

	V1.5.4	2020.05.07	New Command added: CMD_READ_SENSOR_ID_NEW
	1.5.5	2020.06.12	One extra byte battery voltage data added at ITEM_TF_USRch(ch=1…9). #define ITEM_TF_USR1 0x63//Temperature(℃) 4Byte #define ITEM_TF_USR2 0x64//Temperature(℃) 4Byte #define ITEM_TF_USR3 0x65//Temperature(℃) 4Byte #define ITEM_TF_USR4 0x66//Temperature(℃) 4Byte #define ITEM_TF_USR5 0x67//Temperature(℃) 4Byte #define ITEM_TF_USR6 0x68//Temperature(℃) 4Byte #define ITEM_TF_USR7 0x69//Temperature(℃) 4Byte #define ITEM_TF_USR8 0x6A//Temperature(℃) 4Byte
	1.5.6	2020.06.22	Correct typing error： CMD_READ_SSSS = 0x30,//read system info CMD_WRITE_SSSS= 0x31,//write system info
	1.5.7	2020.07.16	Update CMD_READ_SENSOR_ID CMD_READ_SENSOR_ID_NEW
	1.5.8	2020.07.20	Newly added sensor type: #define ITEM_SENSOR_CO2 0x70
	V1.5.9	2020.08.13	CMD_GET_CO2_OFFSET = 0x53, CMD_SET_CO2_OFFSET = 0x54,
	V1.6.0	2021.01.05	With GW1000_Firmware V1.6.5 {leaf wetness sensor} ITEM ("ITEM_LEAF_WETNESS_CHx"(x=1~8)) #define ITEM_LEAF_WETNESS_CH1 0x72 #define ITEM_LEAF_WETNESS_CH2 0x73 #define ITEM_LEAF_WETNESS_CH3 0x74 #define ITEM_LEAF_WETNESS_CH4 0x75 #define ITEM_LEAF_WETNESS_CH5 0x76 #define ITEM_LEAF_WETNESS_CH6 0x77 #define ITEM_LEAF_WETNESS_CH7 0x78 #define ITEM_LEAF_WETNESS_CH8 0x79
	V1.6.1	2021.8.17	Correct CMD_READ_SENSOR_New description error for sensor voltage
	V1.6.2	2021.11.4	Define WS90sensor id. CMD_READ_RSTRAIN_TIME = 0x55, CMD_WRITE_RSTRAIN_TIME = 0x56,
	V1.6.3	2021.11.4	Change wn34 sensor data size from 4 byte to 3 byte

1. Data exchange format：

Fixed header, CMD, SIZE, DATA1, DATA2, … , DATAn, CHECKSUM

Fixed header: 2 bytes, header is fixed as 0xffff

CMD: 1 byte, Command

SIZE: 1 byte, packet size，counted from CMD till CHECKSUM

DATA: n bytes, payloads，variable length

CHECKSUM: 1 byte, CHECKSUM=CMD+SIZE+DATA1+DATA2+…+DATAn

2. Command and data structure definition:

```
typedef  enum  {
//
    CMD_WRITE_SSID              = 0x11,// send SSID and Password to WIFI module
    CMD_BROADCAST               = 0x12,// UDP cast for device echo，answer back data size is 2 Bytes

    CMD_READ_ECOWITT            = 0x1E,// read aw.net setting
    CMD_WRITE_ ECOWITT          = 0x1F, //write back awt.net setting
    CMD_READ_WUNDERGROUND       = 0x20,// read Wunderground setting
    CMD_WRITE_WUNDERGROUND      = 0x21, //write back Wunderground setting
    CMD_READ_WOW                = 0x22, // read WeatherObservationsWebsite setting
    CMD_WRITE_WOW               = 0x23, // write back WeatherObservationsWebsite setting
    CMD_READ_WEATHERCLOUD       = 0x24,// read Weathercloud setting
    CMD_WRITE_WEATHERCLOUD      = 0x25, //write back Weathercloud setting
    CMD_READ_SATION_MAC         = 0x26,// read MAC address
    CMD_READ_CUSTOMIZED         = 0x2A,// read Customized sever setting
    CMD_WRITE_CUSTOMIZED        = 0x2B, // write back Customized sever setting
    CMD_WRITE_UPDATE            = 0x43,// firmware upgrade
    CMD_READ_FIRMWARE_VERSION   = 0x50,// read current firmware version number
    CMD_READ_USR_PATH           = 0x51,
    CMD_WRITE_USR_PATH          = 0x52,

    // the following command is only valid for GW1000, WH2650 and wn1900 支持：
    CMD_GW1000_LIVEDATA         = 0x27, // read current data，reply data size is 2bytes.
    CMD_GET_SOILHUMIAD          = 0x28,// read Soilmoisture Sensor calibration parameters
    CMD_SET_SOILHUMIAD          = 0x29, // write back Soilmoisture Sensor calibration parameters
    CMD_GET_MulCH_OFFSET        = 0x2C, // read multi channel sensor offset value
    CMD_SET_MulCH_OFFSET        = 0x2D, // write back   multi channel sensor OFFSET value
    CMD_GET_PM25_OFFSET         = 0x2E, // read PM2.5OFFSET calibration data
    CMD_SET_PM25_OFFSET         = 0x2F, // writeback PM2.5OFFSET calibration data
    CMD_READ_SSSS               = 0x30,// read system info
    CMD_WRITE_SSSS              = 0x31,// write back system info
    CMD_READ_RAINDATA           = 0x34,// read rain data
    CMD_WRITE_RAINDATA          = 0x35, // write back rain data
    CMD_READ_GAIN               = 0x36, // read rain gain
```

```
        CMD_WRITE_GAIN                  = 0x37, // write back rain gain
        CMD_READ_CALIBRATION            = 0x38,// read sensor set offset calibration value
        CMD_WRITE_CALIBRATION           = 0x39,// write back   sensor set offset value
        CMD_READ_SENSOR_ID              = 0x3A,// read Sensors ID
        CMD_WRITE_SENSOR_ID             = 0x3B, // write back Sensors ID
        CMD_READ_SENSOR_ID_NEW          = 0x3C,//// this is reserved for newly added sensors
        CMD_WRITE_REBOOT                = 0x40,// system restart
        CMD_WRITE_RESET                 = 0x41,// reset to default
       CMD_READ_CUSTOMIZED_PATH         = 0x51,
       CMD_WRITE_CUSTOMIZED_PATH        = 0x52,

       CMD_GET_CO2_OFFSET               = 0x53,//读取 CO2 OFFSET 参数
       CMD_SET_CO2_OFFSET               = 0x54,//改写 CO2 OFFSET 参数

        CMD_READ_RSTRAIN_TIME           = 0x55,// read rain reset time
        CMD_WRITE_RSTRAIN_TIME          = 0x56,// write back rain reset time

      CMD_LIST_UNKNOW,
}CMD_LT;
//*********************************************************************************************
#define SOIL_CH_MAX                     8
#define WH31_CHANNEL                    8
#define PM25_CH_MAX                     4
#define LEAK_CH_MAX                     4
#define LEAF_CH_MAX                     8
typedef enum
{
      //eWH24_SENSOR = 0x00,
      eWH65_SENSOR = 0x00,
      //eWH69_SENSOR,
      eWH68_SENSOR,
      eWH80_SENSOR,   //80H（
      eWH40_SENSOR,
      eWH25_SENSOR,
      eWH26_SENSOR,

      eWH31_SENSORCH1,
      eWH31_SENSORCH2,
      eWH31_SENSORCH3,
      eWH31_SENSORCH4,
      eWH31_SENSORCH5,
      eWH31_SENSORCH6,
      eWH31_SENSORCH7,
```

```
    eWH31_SENSORCH8,
    eWH51_SENSORCH1,
    eWH51_SENSORCH2,
    eWH51_SENSORCH3,
    eWH51_SENSORCH4,
    eWH51_SENSORCH5,
    eWH51_SENSORCH6,
    eWH51_SENSORCH7,
    eWH51_SENSORCH8,
    eWH41_SENSORCH1,
    eWH41_SENSORCH2,
    eWH41_SENSORCH3,
    eWH41_SENSORCH4,
    //-------------
    eWH57_SENSOR,
    eWH55_SENSORCH1,
    eWH55_SENSORCH2,
    eWH55_SENSORCH3,
    eWH55_SENSORCH4,
    eWH34_SENSORCH1 = 31,
    eWH34_SENSORCH2 = 32,
    eWH34_SENSORCH3 = 33,
    eWH34_SENSORCH4 = 34,
    eWH34_SENSORCH5 = 35,
    eWH34_SENSORCH6 = 36,
    eWH34_SENSORCH7 = 37,
    eWH34_SENSORCH8 = 38,
      eWH45_SENSOR = 39,
      // GW1000 Firmware V1.5.6 之后加入的
      eWH35_SENSORCH1 = 40,
      eWH35_SENSORCH2 = 41,
      eWH35_SENSORCH3 = 42,
      eWH35_SENSORCH4 = 43,
      eWH35_SENSORCH5 = 44,
      eWH35_SENSORCH6 = 45,
      eWH35_SENSORCH7 = 46,
      eWH35_SENSORCH8 = 47,
     eWH90_SENSOR = 48,

    // the sensor sequence can not be altered!!
    //-------------

    eMAX_SENSOR
```

```
}SENSOR_IDT;
//-------------------------------------------------------------------------------------------------------------------------------
#define ITEM_INTEMP             0x01//Indoor Temperature (℃)        2
#define ITEM_OUTTEMP            0x02//Outdoor Temperature (℃)       2
#define ITEM_DEWPOINT           0x03//Dew point (℃)                 2
#define ITEM_WINDCHILL          0x04//Wind chill (℃)                2
#define ITEM_HEATINDEX          0x05//Heat index (℃)                2
#define ITEM_INHUMI             0x06//Indoor Humidity (%)           1
#define ITEM_OUTHUMI            0x07//Outdoor Humidity (%)          1
#define ITEM_ABSBARO            0x08//Absolutely Barometric (hpa)   2
#define ITEM_RELBARO            0x09//Relative Barometric (hpa)     2
#define ITEM_WINDDIRECTION      0x0A//Wind Direction (360°)         2
#define ITEM_WINDSPEED          0x0B//Wind Speed (m/s)              2
#define ITEM_GUSTSPEED          0x0C//Gust Speed (m/s)              2
#define ITEM_RAINEVENT          0x0D//Rain Event (mm)               2
#define ITEM_RAINRATE           0x0E//Rain Rate (mm/h)              2
#define ITEM_RAINHOUR           0x0F//Rain hour (mm)                2
#define ITEM_RAINDAY            0x10//Rain Day (mm)                 2
#define ITEM_RAINWEEK           0x11//Rain Week (mm)                2
#define ITEM_RAINMONTH          0x12//Rain Month (mm)               4
#define ITEM_RAINYEAR           0x13//Rain Year (mm)                4
#define ITEM_RAINTOTALS         0x14//Rain Totals (mm)              4
#define ITEM_LIGHT              0x15//Light (lux)                   4
#define ITEM_UV                 0x16//UV (uW/m2)                   2
#define ITEM_UVI                0x17//UVI (0-15 index)             1
#define ITEM_TIME               0x18//Date and time                 6
#define ITEM_DAYLWINDMAX        0X19//Day max wind(m/s)             2
#define ITEM_TEMP1              0x1A//Temperature 1(℃)              2
#define ITEM_TEMP2              0x1B//Temperature 2(℃)              2
#define ITEM_TEMP3              0x1C//Temperature 3(℃)              2
#define ITEM_TEMP4              0x1D//Temperature 4(℃)              2
#define ITEM_TEMP5              0x1E//Temperature 5(℃)              2
#define ITEM_TEMP6              0x1F//Temperature 6(℃)              2
#define ITEM_TEMP7              0x20//Temperature 7(℃)              2
#define ITEM_TEMP8              0x21//Temperature 8(℃)              2
#define ITEM_HUMI1              0x22//Humidity 1, 0-100%            1
#define ITEM_HUMI2              0x23//Humidity 2, 0-100%            1
#define ITEM_HUMI3              0x24//Humidity 3, 0-100%            1
#define ITEM_HUMI4              0x25//Humidity 4, 0-100%            1
#define ITEM_HUMI5              0x26//Humidity 5, 0-100%            1
#define ITEM_HUMI6              0x27//Humidity 6, 0-100%            1
#define ITEM_HUMI7              0x28//Humidity 7, 0-100%            1
#define ITEM_HUMI8              0x29//Humidity 8, 0-100%            1
```

#define ITEM_PM25_CH1	0x2A//PM2.5 Air Quality Sensor(μg/m3)	2
#define ITEM_SOILTEMP1	0x2B//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE1	0x2C//Soil Moisture(%)	1
#define ITEM_SOILTEMP2	0x2D//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE2	0x2E//Soil Moisture(%)	1
#define ITEM_SOILTEMP3	0x2F//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE3	0x30//Soil Moisture(%)	1
#define ITEM_SOILTEMP4	0x31//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE4	0x32//Soil Moisture(%)	1
#define ITEM_SOILTEMP5	0x33//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE5	0x34//Soil Moisture(%)	1
#define ITEM_SOILTEMP6	0x35//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE6	0x36//Soil Moisture(%)	1
#define ITEM_SOILTEMP7	0x37//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE7	0x38//Soil Moisture(%)	1
#define ITEM_SOILTEMP8	0x39//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE8	0x3A//Soil Moisture(%)	1
#define ITEM_SOILTEMP9	0x3B//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE9	0x3C//Soil Moisture(%)	1
#define ITEM_SOILTEMP10	0x3D//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE10	0x3E//Soil Moisture(%)	1
#define ITEM_SOILTEMP11	0x3F//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE11	0x40//Soil Moisture(%)	1
#define ITEM_SOILTEMP12	0x41//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE12	0x42//Soil Moisture(%)	1
#define ITEM_SOILTEMP13	0x43//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE13	0x44//Soil Moisture(%)	1
#define ITEM_SOILTEMP14	0x45//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE14	0x46//Soil Moisture(%)	1
#define ITEM_SOILTEMP15	0x47//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE15	0x48//Soil Moisture(%)	1
#define ITEM_SOILTEMP16	0x49//Soil Temperature(℃)	2
#define ITEM_SOILMOISTURE16	0x4A//Soil Moisture(%)	1
#define ITEM_LOWBATT	0x4C//All sensor lowbatt 16 char	16
#define ITEM_PM25_24HAVG1	0x4D//for pm25_ch1	2
#define ITEM_PM25_24HAVG2	0x4E//for pm25_ch2	2
#define ITEM_PM25_24HAVG3	0x4F//for pm25_ch3	2
#define ITEM_PM25_24HAVG4	0x50//for pm25_ch4	2
#define ITEM_PM25_CH2	0x51//PM2.5 Air Quality Sensor(μg/m3)	2
#define ITEM_PM25_CH3	0x52//PM2.5 Air Quality Sensor(μg/m3)	2

```
#define ITEM_PM25_CH4                   0x53//PM2.5 Air Quality Sensor(μg/m3)          2

#define ITEM_LEAK_CH1                   0x58//for Leak_ch1                             1
#define ITEM_LEAK_CH2                   0x59//for Leak_ch2                             1
#define ITEM_LEAK_CH3                   0x5A//for Leak_ch3                             1
#define ITEM_LEAK_CH4                   0x5B//for Leak_ch4                             1

#define ITEM_LIGHTNING                  0x60 // lightning distance （1~40KM）          1
#define ITEM_LIGHTNING_TIME             0x61// lightning happened time(UTC)           4
#define ITEM_LIGHTNING_POWER            0x62// lightning counter for the ay           4

#define ITEM_TF_USR1                    0x63//Temperature(℃)                           3
#define ITEM_TF_USR2                    0x64//Temperature(℃)                           3
#define ITEM_TF_USR3                    0x65//Temperature(℃)                           3
#define ITEM_TF_USR4                    0x66//Temperature(℃)                           3
#define ITEM_TF_USR5                    0x67//Temperature(℃)                           3
#define ITEM_TF_USR6                    0x68//Temperature(℃)                           3
#define ITEM_TF_USR7                    0x69//Temperature(℃)                           3
#define ITEM_TF_USR8                    0x6A//Temperature(℃)                           3

//the data packet in this sequence, and it should be followed strictly this data sequence
#define ITEM_SENSOR_CO2                 0x70                                           16
/* ------------------Ecowitt-----------------
   1   tf_co2                short                    C           x10
   2   humi_co2              unsigned char            %
   3   pm10_co2              unsigned short           ug/m3       x10
   4   pm10_24h_co2          unsigned short           ug/m3       x10
   5   pm25_co2              unsigned short           ug/m3       x10
   6   pm25_24h_co2          unsigned short           ug/m3       x10
   7   co2                   unsigned short           ppm
   8   co2_24h               unsigned short           ppm
   9   co2_batt              u8                       (0~5)
   ------------------------------------------- */
//-----------------------------------------------------------------------------------------------------------------------
#define ITEM_PM25_AQI                   0x71 //only for amb
// ITEM_PM25_AQI   length(n*2)(1byte)   1-aqi_pm25 2-aqi_pm25_24h ... ... n-aqi
/*
aqi_pm25                AQI derived from PM25   int
aqi_pm25_24h            AQI derived from PM25, 24 hour running average   int
aqi_pm25_in             AQI derived from PM25 IN   int
aqi_pm25_in_24h         AQI derived from PM25 IN, 24 hour running average int
aqi_pm25_aqin           AQI derived from PM25, AQIN sensor   int
aqi_pm25_24h_aqin       AQI derived from PM25, 24 hour running average, AQIN sensor int
```

```
.... n
*/
//-----------------------------------------------------------------------------------------------------------------------
#define ITEM_LEAF_WETNESS_CH1     0x72//                                    1
#define ITEM_LEAF_WETNESS_CH2     0x73//                                    1
#define ITEM_LEAF_WETNESS_CH3     0x74//                                    1
#define ITEM_LEAF_WETNESS_CH4     0x75//                                    1
#define ITEM_LEAF_WETNESS_CH5     0x76//                                    1
#define ITEM_LEAF_WETNESS_CH6     0x77//                                    1
#define ITEM_LEAF_WETNESS_CH7     0x78//                                    1
#define ITEM_LEAF_WETNESS_CH8     0x79//                                    1
//-----------------------------------------------------------------------------------------------------------------------

//-----------------------------------------------------------------------------------------------------------------------
#if 0 // GW1000 Firmware V1.6.5 or after stop using this entity.   CMD_READ_SENSOR_ID_NEW interpret each sensor for battery status .
#if 1
typedef union //1 low bat   0 bat normal
{
    unsigned char batt;
    struct
    {
        unsigned char wh41      : 4; /* bit 0~3 */ // 0~5
        unsigned char wh40      : 1; /* bit 4 */
        unsigned char wh26      : 1; /* bit 5 */
        unsigned char wh25      : 1; /* bit 6 */
        unsigned char wh24      : 1; /* bit 7 */ //65, 69
    } bits;
} _sig_sen;

typedef union //1 low batt   0 batt normal
{
    unsigned char batt;
    Struct {
        unsigned char   ch1   : 1;      /* bit 0 */
        unsigned char   ch2   : 1;      /* bit 1 */
        unsigned char   ch3   : 1;      /* bit 2 */
        unsigned char   ch4   : 1;      /* bit 3 */
        unsigned char   ch5   : 1;      /* bit 4 */
        unsigned char   ch6   : 1;      /* bit 5 */
        unsigned char   ch7   : 1;      /* bit 6 */
        unsigned char   ch8   : 1;/* bit 7 */
    } bits;
```

```
} _wh31_ch;

typedef union //val
{
    unsigned short batt;
    struct    {
        unsigned char ch1    : 4;/* bit 0~3    */ // 0~5
        unsigned char ch2    : 4;/* bit 4~7    */ // 0~5
        unsigned char ch3    : 4;/* bit 8~11   */ // 0~5
        unsigned char ch4    : 4;/* bit 12~15 */ // 0~5
    } bits;
} _wh41_ch;
typedef union //1 low 0 normal
{
    unsigned short  batt;
    struct    {
        unsigned char    ch1  : 1;/* bit 0 */
        unsigned char    ch2  : 1;      /* bit 1 */
        unsigned char    ch3  : 1;      /* bit 2 */
        unsigned char    ch4  : 1;      /* bit 3 */
        unsigned char    ch5  : 1;      /* bit 4 */
        unsigned char    ch6  : 1;      /* bit 5 */
        unsigned char    ch7  : 1;      /* bit 6 */
        unsigned char    ch8  : 1;/* bit 7 */

        unsigned char    ch9   : 1;     /* bit 8   */
        unsigned char    ch10  : 1;     /* bit 9   */
        unsigned char    ch11  : 1;     /* bit 10 */
        unsigned char    ch12  : 1;     /* bit 11 */
        unsigned char    ch13  : 1;     /* bit 12 */
        unsigned char    ch14  : 1;     /* bit 13 */
        unsigned char    ch15  : 1;     /* bit 14 */
       unsigned char    ch16  : 1;      /* bit 15 */
    } word;
} _wh51_ch;
// battery voltage and low battery correlation
typedef union _sensor_batt
{
    unsigned char all_batt[16];
    // represented with battery voltage
    struct
    {
        _sig_sen single;
```

~~_wh31_ch wh31;~~
~~_wh51_ch wh51;~~
~~unsigned char wh57; // 0~5~~
~~unsigned char wh68; // 0.02V * val(received val) = wh68(current voltage);~~
~~unsigned char wh80; // 0.02V * val(received val) = wh80(current voltage);~~
~~unsigned char wh45; // 0~5~~
~~_wh41_ch wh41; // batter level 0~5, <=1 for low battery~~
~~unsigned char wh55[LEAK_CH_MAX]; //~~
~~} val;~~
~~} sensor_batt;//~~

~~// type AP_SSID~~
~~typedef AP_SSID~~
~~{~~
~~Size byte // ssid size~~
~~SSID n // n max 32~~
~~}~~

~~#endif~~

~~//V1.5.9 introduced:~~
~~unsigned char wh34[TF_CH_MAX];//0.02V * val(received val) = wh34(current voltage);~~
~~//----------~~

3. Wi-Fi configuration and looking for device within the local network

Wifi configuration 1 : APP side use took port 49123 and setup a TCP Server. WIFI module side works with station+AP mode and setup a TCP Client to be connected with APP side server. If WIFI module connected successfully to APP side TCP server, APP side TCP Server send CMD_WRITE_SSID command!

Wifi configuration 2: WIFI module works at station+AP mode, and setup a TCP Server at port 45000, waiting APP side for connection. When APP side connected to WIFI moule's TCP Server，CMD_WRITE_SSID can be issued.

Description	Length（Bytes）	Note
Fixed header	2	Fixed as 0xffff
CMD_WRITE_SSID	1	0x11
Size	1	Packet size
SSID Size	1	SSID length
SSID	n	Max 32
Password Size	1	Password length
Password	n	Max 64
Checksum	1	Checksum

WIFI answer back：

description	length（byte）	notes
Fixed header	2	Fixed as 0xffff

CMD_WRITE_SSID	1	0x11
Size	1	Packet size
Result	1	0x00: set success，0x01: set fail
Checksum	1	checksum

When console and APP device are on the same WLAN network, connected to a same router, APP side will send a command via UDP casting, after this command received, console wifi will reply back its MAC, IP, Port and AP's SSID to APP device side. (port 46000).

Description	Length	Notes
Fixed header	2	Fixed as 0xffff
CMD_BROADCAST	1	0x12
Size	1	Packet size
Checksum	1	Checksum

WIFI replied：

Description	Length	Notes
Fixed header	2	Fixed as 0xffff
CMD_BROADCAST	1	0x12
Size	2	Packet size (size is 2 Bytes)
MAC	6	Wifi module STA MAC
IP1	1	Eg. 192.168.100.1 中的 192
IP2	1	Eg. 192.168.100.1 中的 168
IP3	1	Eg. 192.168.100.1 中的 100
IP4	1	Eg. 192.168.100.1 中的 1
Port1	1	Eg. 0x1194(45000)中的 0x11
Port2	1	Eg. 0x1194(45000)中的 0x94
AP SSID	n	Wifi module's AP SSID
Checksum	1	checksum

Console WIFI module setup TCP server at port 45000, waiting for APP side to be connected. If connection established, use the following command:

1) Read ambietweather.net setting

Description	Length	Notes
Fixed header	2	Fixed as 0xffff
CMD_READ_ECOWITT	1	0x1E
Size	1	Packet size
Checksum	1	checksum

Console WIFI return：

Description	Length	Notes
Fixed header	2	Fixed as 0xffff
CMD_READ_ECOWITT	1	0x1E
Size	1	Packet size
Uplaod interval	1	0~5min (0: mean is OFF)

Checksum	1	checksum

2) Rewrite ambientweatehr.net setting

Description	Length	Notes
Fixed header	2	Fixed as 0xffff
CMD_WRITE_ECOWITT	1	0x1F
Size	1	Packet size
Uplaod interval	1	0~5min (0: mean is OFF)
Checksum	1	checksum

Console WIFI module return data:

Description	Length	Notes
Fixed header	2	Fixed as 0xffff
CMD_WRITE_ECOWITT	1	0x1F
Size	1	Packet size
Result	1	0x00:success，0x01: fail
Checksum	1	checksum

3) Read back Wunderground setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_WUNDERGROUND	1	0x20
Size	1	Packet size
Checksum	1	checksum

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_WUNDERGROUND	1	0x20
Size	1	Packet size
ID Size	1	ID Size
ID	n	ASCII , max 32
Password Size	1	Password Size
Password	n	ASCII , max 32
Fix	1	1
Checksum	1	checksum

4) Rewrite Wunderground setting

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_WUNDERGROUND	1	0x21
Size	1	Packet size
ID Size	1	ID Size
ID	n	ASCII , max 32

Password Size	1	Password Size
Password	n	ASCII , max 32
Fix	1	1
Checksum	1	checksum

Console WIFI module return data：

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_WUNDERGROUND	1	0x21
Size	1	Packet size
Result	1	0x00:sucess，0x01: fail
Checksum	1	checksum

5) Read WeatherObservationsWebsite setting

Description	Length	Notes
Fixed header	2	固定 0xffff
CMD_READ_WOW	1	0x22
Size	1	包长度
Checksum	1	checksum

Console return：

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_WOW	1	0x22
Size	1	Packet size
ID Size	1	ID Size
ID	n	ASCII , max 39
Password Size	1	Password Size
Password	n	ASCII , max 32
StationNum Size（unused）	1	StationNum size（unused）
StationNum（unused）	n	ASCII , max 32（unused）
Fix	1	1
Checksum	1	checksum

6) Rewrite WeatherObservationsWebsite setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_WOW	1	0x23
Size	1	Packet size
ID Size	1	ID Size
ID	n	ASCII , max 39
Password Size	1	Password Size
Password	n	ASCII , max 32
StationNum Size（unused）	1	StationNum size（unused）

StationNum（unused）	32	ASCII , max 32（unused）
Fix	1	1
Checksum	1	checksum

Console return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_WOW	1	0x23
Size	1	Packet size
Result	1	0x00:sucess，0x01: fail
Checksum	1	checksum

7) Read Weathercloud setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_WEATHERCLOUD	1	0x24
Size	1	
Checksum	1	checksum

Console return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_WEATHERCLOUD	1	0x24
Size	1	
ID Size	1	ID Size
ID	n	ASCII , max 32
Key Size	1	Key Size
Key	n	ASCII , max 32
Fix	1	1
Checksum	1	checksum

8) Write back Weathercloud setting

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_WEATHERCLOUD	1	0x25
Size	1	
ID Size	1	ID Size
ID	n	ASCII , max 32
Key Size	1	Key Size
Key	n	ASCII , max 32
Fix	1	1
Checksum	1	checksum

Console return：

Description	Length	Notes

Fixed header	2	0xffff
CMD_WRITE_WEATHERCLOUD	1	0x25
Size	1	
Result	1	0x00: success，0x01: fail
Checksum	1	checksum

9) Read customer server setting

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_CUSTOMIZED	1	0x2A
Size	1	
Checksum	1	checksum

Console return :

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_CUSTOMIZED	1	0x2A
Size	1	
ID Size	1	ID Size
ID	n	ASCII , max 40
Password Size	1	Password Size
Password	n	ASCII , max 40
Server Size	1	Server Size
Server	n	ASCII , max 64
Port	2	0-65535
Interval	2	16-600
Type	1	0:EC 1WU
Active	1	0:Disable 1:Enable
Checksum	1	checksum

10) Write back customer server setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_CUSTOMIZED	1	0x2B
Size	1	
ID Size	1	ID Size
ID	n	ASCII , max 40
Password Size	1	Password Size
Password	n	ASCII , max 40
Server Size	1	Server Size
Server	n	ASCII , max 64
Port	2	0-65535
Interval	2	16-600

Type	1	0:EC 1WU
Active	1	0:Disable 1:Enable
Checksum	1	checksum

Console WiFi return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_CUSTOMIZED	1	0x2B
Size	1	
Result	1	0x00:sucess，0x01: fail
Checksum	1	checksum

11) Read customer usr_path setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_USRPATH	1	0x51
Size	1	
Checksum	1	checksum

Console WIFI return：

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_USRPATH	1	0x51
Size	1	
Ecowitt Path	64	ASCII , max 64
WU Path	64	ASCII , max 64
Checksum	1	checksum

12) Write back customer usr_path setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_USRPATH	1	0x52
Size	1	
Ecowitt Path	64	ASCII , max 64
WU Path	64	ASCII , max 64
Checksum	1	checksum

Console WIFI return：

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_USRPATH	1	0x52
Size	1	
Result	1	0x00:sucess，0x01: fail
Checksum	1	checksum

13) Read Soilmoisture Sensor calibration parameter:

Description	Length	Notes
Fixed header	2	0xffff
CMD_GET_SOILHUMIAD	1	0x28
Size	1	
Checksum	1	checksum

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_GET_SOILHUMIAD	1	0x29
Size	1	
Channel	1	Channel number
Current humidity	1	Send by sensor
Current ad	2	Send by sensor
Customize Calibration Option	1	=1 enable, =0 default by sensor
Min ad	1	Customize Mode 0% AD(70~200)
Max ad	2	Customize Mode 100% AD(80~1000)
...	...	
Checksum	1	checksum

14) Write back Soilmoisture Sensor calibration setting

Description	Length	Notes
Fixed header	2	0xffff
CMD_SET_SOILHUMIAD	1	0x29
Size	1	
Channel	1	Channel number
Customize Calibration Option	1	=1 enable, otherwise =0
Min ad	1	Customize Mode 0% ADd(70~200)
Max ad	2	Customize Mode 100% ad(80~1000)
...	...	
Checksum	1	checksum

Console WIFI return：

Description	Length	Notes
Fixed header	2	0xffff
CMD_SET_SOILHUMIAD	1	0x29
Size	1	
Result	1	0x00: sucess，0x01: fail
Checksum	1	checksum

15) Read multi channel temp sensor OFFSET setting

Description	Length	Notes

Fixed header	2	0xffff
CMD_GET_MulCH_OFFSET	1	0x2C
Size	1	
Checksum	1	checksum

Console return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_GET_MulCH_OFFSET	1	0x2C
Size	1	
Channel	1	
humidity offset	1	Range: -10 ~ 10, default: 0
Temperature offset	1	Range: -100~100, default: 0 Note: (-10.0℃~10.0℃)x10
.....	....	
WH31_CHANNEL-1	1	0~7
humidity offset	1	
Temperature offset	1	Range: -100~100, default: 0 Note: (-10.0℃~10.0℃)x10
Checksum	1	checksum

16) Write back multi channel temp sensor OFFSET setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_SET_MulCH_OFFSET	1	0x2D
Size	1	
Channel	1	
humidity offset	1	Range: -10 ~ 10, default: 0
Temperature offset	1	Range: -100~100, default: 0 Note: (-10.0℃~10.0℃)x10
.....	....	
WH31_CHANNEL-1	1	0~7
humidity offset	1	
Temperature offset	1	Range: -100~100, default: 0 Note: (-10.0℃~10.0℃)x10
Checksum	1	checksum

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_SET_MulCH_OFFSET	1	
Size	1	
Result	1	0x00: success , 0x01: fail
Checksum	1	checksum

17) Read multi channel PM2.5OFFSET calibration setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_GET_PM25_OFFSET	1	0x2E
Size	1	
Checksum	1	checksum

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_GET_PM25_OFFSET	1	0x2E
Size	1	
Channel	1	
PM25offset	2	Range: -200 ~ 200, default: 0 Note: (-20~20 ug/m3)x10
.....	....	
PM25_CH_MAX-1	1	0~3
PM25offset	1	
Checksum	1	checksum

18) Write back multi channel PM2.5OFFSET calibration:

Description	Length	Notes
Fixed header	2	0xffff
CMD_SET_PM25_OFFSET	1	0x2F
Size	1	
Channel	1	
PM25offset	2	Range: -200 ~ 200, default: 0 Note: (-20~20 ug/m3)x10
.....	....	
PM25_CH_MAX-1	1	0~3
PM25offset	1	
Checksum	1	checksum

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_SET_PM25_OFFSET	1	0x2F
Size	1	
Result	1	0x00:sucess，0x01: fail
Checksum	1	checksum

19) Read back CO2 OFFSET calibration

field	bytes	Notes

Fixed header	2	0xffff
CMD_GET_CO2_OFFSET	1	0x53
Size	1	
Checksum	1	checksum

WIFI reply：

Field	Bytes	Notes
Fixed header	2	0xffff
CMD_GET_CO2_OFFSET	1	0x53
Size	1	
CO2 offset	2	Range: -600 ~ 10000, default: 0
PM25offset	2	Range: -200 ~ 200, default: 0 Note: (-20~20 ug/m3)x10
PM10offset	2	Range: -200 ~ 200, default: 0 Note: (-20~20 ug/m3)x10
Checksum	1	checksum

20) Update CO2 OFFSET calibration

field	bytes	Note
Fixed header	2	fixed 0xffff
CMD_SET_CO2_OFFSET	1	0x54
Size	1	Packet size
CO2 offset	2	Range: -600 ~ 10000, default: 0
PM25offset	2	Range: -200 ~ 200, default: 0 Note: (-20~20 ug/m3)x10
PM10offset	2	Range: -200 ~ 200, default: 0 Note: (-20~20 ug/m3)x10
Checksum	1	checksum

WIFI return：

field	bytes	notes
Fixed header	2	Fixed 0xffff
CMD_SET_CO2_OFFSET	1	0x54
Size	1	Packet size
Result	1	0x00:success，0x01: fail
Checksum	1	checksum

21)读取 Rain RST TIME

字 段	长度（字节）	说 明
Fixed header	2	固定 0xffff
CMD_READ_RSTRAIN_TIME	1	0x55
Size	1	包长度
Checksum	1	checksum

WIFI 模块返回数据：

字 段	长度（字节）	说 明
Fixed header	2	固定 0xffff
CMD_READ_RSTRAIN_TIME	1	0x55
Size	1	包长度
rstRainDayTime	1	Range: 0 ~ 23, default: 0
rstRainWeekTime	1	Range: 0 or 1, default: 0 0: Sunday, 1:Monday;
rstRainYearTime	1	Range: 0 ~ 11 , default: 0 0:January ~ 11:December
Checksum	1	checksum

22)改写 Rain RST TIME

字 段	长度（字节）	说 明
Fixed header	2	固定 0xffff
CMD_WRITE_RSTRAIN_TIME	1	0x56
Size	1	包长度
rstRainDayTime	1	Range: 0 ~ 23, default: 0
rstRainWeekTime	1	Range: 0 or 1, default: 0 0: Sunday, 1:Monday;
rstRainYearTime	1	Range: 0 ~ 11 , default: 0 0:January ~ 11:December
Checksum	1	checksum

WIFI 模块返回数据：

字 段	长度（字节）	说 明
Fixed header	2	固定 0xffff
CMD_WRITE_RSTRAIN_TIME	1	0x56
Size	1	包长度
Result	1	0x00:成功，0x01: 失败
Checksum	1	checksum

23) Read MAC

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_SATION_MAC	1	0x26
Size	1	
Result	1	0x00:sucess，0x01: fail
Checksum	1	checksum

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_SATION_MAC	1	0x26

Size	1	
Sta_mac[6]	6	sta_mac[0];sta_mac[1];sta_mac[2]; sta_mac[3];sta_mac[4];sta_mac[5];
Checksum	1	checksum

24) Read current sensor data. Note: returned packet size for data payload is 2 bytes.

Description	Length	Notes
Fixed header	2	0xffff
CMD_GW1000_LIVEDATA	1	0x27
Size	1	
Checksum	1	checksum

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_GW1000_LIVEDATA	1	0x27
Size	2	Size is2 Byte
ITEM_PM25	1	
Value	2	Unsigned short (valuex10)
ITEM_PM10	1	
Value	2	Unsigned short (valuex10)
ITEM_CH1_SOil_H	1	
Value	1	0~99
ITEM_CH2_TEMP	1	
Value	2	signed short (valuex10)
... ...		
ITEM_CH7_TEMP	1	
Value	2	
ITEM_CH1_HUMI	1	
Value	1	0~99
ITEM_CH2_HUMI	1	
ITEM_LOWBATT	1	
Value	16	typedef union _sensor_batt
... ...		
ITEM_CH7_HUMI	1	
Value	1	
ITEM_TF_USR1	1	
Temperature Value	2	signed short (valuex10)
Battery Value	1	0.02V * val
.....	.....	
ITEM_TF_USR8	1	
Temperature Value	2	signed short (valuex10)

Battery Value	1	0.02V * val
.....	.....	
ITEM_SENSOR_CO2	1	
tf_co2 value	2	signed short (valuex10)
humi_co2 value	1	
pm10_co2 value	2	unsigned short(valuex10)
pm10_24h_co2 value	2	unsigned short(valuex10)
pm25_co2 value	2	unsigned short(valuex10)
pm25_24h_co2 value	2	unsigned short(valuex10)
Co2 value	2	unsigned short
co2_24h value	2	unsigned short
co2_batt value	1	0～5
Checksum	1	checksum

25) Read system parameter:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_SSSS	1	0x30
Size	1	
Checksum	1	checksum

WIFI 模块返回数据：

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_SSSS	1	0x30
Size	1	
Frequency (注)	1	Wireless Receive Frequency(Only read)
Sensor Type	1	0:WH24 1:WH65
UTC TIME	4	Unsigned long(Only read)
Timezone Index	1	Local time zone index
DST Status	1	True or False
Checksum	1	checksum

```
typedef enum
{
    RFM433M = (unsigned char) 0,//  433MHz
    RFM868M = (unsigned char) 1,//  868MHz
    RFM915M = (unsigned char) 2,//  915MHz
    RFM920M = (unsigned char) 3 //  920MHz
} freq_t;
extern freq_t  Frequency;
```

26) Write back system parameter

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_SSSS	1	0x31
Size	1	
Frequency	1	(Only read)Can't be rewritten.
Sensor Type	1	0:WH24 1:WH65
UTC TIME	4	Unsigned long(Only read)
Timezone Index	1	Local time zone index
DST Status	1	True or False
Checksum	1	checksum

Console WIFI return

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_SSSS	1	0x31
Size	1	
Result	1	0x00:sucess，0x01: fail
Checksum	1	checksum

27) Read rainfall data

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_RAINDATA	1	0x34
Size	1	
Checksum	1	checksum

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_RAINDATA	1	0x34
Size	1	
RainRate	4	Range: 0~60000 Note: (0mm ~6000.0mm)x10
RainDay	4	Range: 0~99999 Note: (0mm ~9999.9mm)x10
RainWeek	4	Range: 0~99999 Note: (0mm ~9999.9mm)x10
RainMonth	4	Range: 0~99999 Note: (0mm ~9999.9mm)x10
RainYear	4	Range: 0~99999 Note: (0mm ~9999.9mm)x10
Checksum	1	

Write back rainfall:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_RAINDATA	1	0x35
Size	1	
RainDay	4	Range: 0~99999 Note: (0mm ~9999.9mm)x10
RainWeek	4	Range: 0~99999 Note: (0mm ~9999.9mm)x10
RainMonth	4	Range: 0~99999 Note: (0mm ~9999.9mm)x10
RainYear	4	Range: 0~99999 Note: (0mm ~9999.9mm)x10
Checksum	1	

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_RAINDATA	1	0x35
Size	1	
Resulte	1	0x00:sucess，0x01: fail
Checksum	1	checksum

28) Read sensor array calibration

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_GAIN	1	0x36
Size	1	
Checksum	1	checksum

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_GAIN	1	0x36
Size	1	
Fixed	2	1267
uvGain	2	Range: 10~500, default: 100 Note: (0.10 ~5.00)x100
solarRadGain	2	Range: 10~500, default: 100 Note: (0.10 ~5.00)x100
windGain	2	Range: 10~500, default: 100 Note: (0.10 ~5.00)x100
rainGain	2	Range: 10~500, default: 100 Note: (0.10 ~5.00)x100

Reserved	2	Reserved
Checksum	1	

Write back calibration setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_ GAIN	1	0x37
Size	1	
Reserved	2	1267(x 10)
uvGain	2	Range: 10~500, default: 100 Note: (0.10 ~5.00)x100
solarRadGain	2	Range: 10~500, default: 100 Note: (0.10 ~5.00)x100
windGain	2	Range: 10~500, default: 100 Note: (0.10 ~5.00)x100
rainGain	2	Range: 10~500, default: 100 Note: (0.10 ~5.00)x100
Reserved	2	Reserved
Checksum	1	

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_ GAIN	1	0x37
Size	1	
Resulte	1	0x00:sucess，0x01: fail
Checksum	1	checksum

29) Read sensor array offset setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_CALIBRATION	1	0x38
Size	1	
Checksum	1	checksum

Console WIFI return：

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_CALIBRATION	1	0x38
Size	1	
inTempOffset	2	Range: -100~100, default: 0 Note: (-10.0℃~10.0℃)x10
inHumiOffset	1	Range: -10~10, default: 0
AbsOffset	4	Range: -800~800, default: 0

		Note: (-80.0hpa~80.0hpa)x10
RelOffset	4	Range: -800~800, default: 0 Note: (-80.0hpa~80.0hpa)x10
outTempOffset	2	Range: -100~100, default: 0 Note: (-10.0℃~10.0℃)x10
outHumiOffset	1	Range: -10~10, default: 0
windDirOffset	2	Range: -180~180, default: 0
Checksum	1	

30) Write back array offset setting:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_CALIBRATION	1	0x39
Size	1	
inTempOffset	2	Range: -100~100, default: 0 Note: (-10.0℃~10.0℃)x10
inHumiOffset	1	Range: -10~10, default: 0
AbsOffset	4	Range: -800~800, default: 0 Note: (-80.0hpa~80.0hpa)x10
RelOffset	4	Range: -800~800, default: 0 Note: (-80.0hpa~80.0hpa)x10
outTempOffset	2	Range: -100~100, default: 0 Note: (-10.0℃~10.0℃)x10
outHumiOffset	1	Range: -10~10, default: 0
windDirOffset	2	Range: -180~180, default: 0
Checksum	1	

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_CALIBRATION	1	0x39
Size	1	
Resulte	1	0x00:sucess，0x01: fail
Checksum	1	checksum

31) Read Sensors ID parameter:

```
typedef enum
{
    //eWH24_SENSOR = 0x00,
    eWH65_SENSOR = 0x00,//   1: BATT low, 0: normal
    //eWH69_SENSOR,
    eWH68_SENSOR,// voltage=val*0.02V   if <=1.2V BAT is low
    eWH80_SENSOR,// 0.02V * val(received val) = wh80;
```

eWH40_SENSOR,// 0.1v * val
eWH25_SENSOR,// 1: BATT low, 0: normal
eWH26_SENSOR,// 1: BATT low, 0: normal

eWH31_SENSORCH1,// 1: BATT low, 0: normal
eWH31_SENSORCH2,// 1: BATT low, 0: normal
eWH31_SENSORCH3,// 1: BATT low, 0: normal
eWH31_SENSORCH4,// 1: BATT low, 0: normal
eWH31_SENSORCH5,// 1: BATT low, 0: normal
eWH31_SENSORCH6,// 1: BATT low, 0: normal
eWH31_SENSORCH7,// 1: BATT low, 0: normal
eWH31_SENSORCH8,// 1: BATT low, 0: normal
eWH51_SENSORCH1,// val * 0.1v
eWH51_SENSORCH2,// val * 0.1
eWH51_SENSORCH3,// val * 0.1
eWH51_SENSORCH4,// val * 0.1
eWH51_SENSORCH5,// val * 0.1
eWH51_SENSORCH6,// val * 0.1
eWH51_SENSORCH7,// val * 0.1
eWH51_SENSORCH8,// val * 0.1
eWH41_SENSORCH1,// level0~6，<=1 for BATT low, 6 = dc power supply
eWH41_SENSORCH2,// level0~6，<=1 for BATT low 6 = dc power supply
eWH41_SENSORCH3,// level0~6，<=1 for BATT low 6 = dc power supply
eWH41_SENSORCH4,// level0~6，<=1 for BATT low 6 = dc power supply
//----------
eWH57_SENSOR, // level0~5，<=1 for BATT low
eWH55_SENSORCH1,// level0~5，<=1 for BATT low
eWH55_SENSORCH2,// level0~5，<=1 for BATT low
eWH55_SENSORCH3,// level0~5，<=1 for BATT low
eWH55_SENSORCH4,// level0~5，<=1 for BATT low
eWH34_SENSORCH1 = 31,// v=val*0.02V if v<=1.2V BATT low
eWH34_SENSORCH2 = 32,// v=val*0.02V if v<=1.2V BATT low
eWH34_SENSORCH3 = 33,// v=val*0.02V if v<=1.2V BATT low
eWH34_SENSORCH4 = 34,// v=val*0.02V if v<=1.2V BATT low
eWH34_SENSORCH5 = 35,// v=val*0.02V if v<=1.2V BATT low
eWH34_SENSORCH6 = 36,// v=val*0.02V if v<=1.2V BATT low
eWH34_SENSORCH7 = 37,// v=val*0.02V if v<=1.2V BATT low
eWH34_SENSORCH8 = 38,// v=val*0.02V if v<=1.2V BATT low
eWH45_SENSOR = 39,// 0~5 6 = dc power supply
// command 0x3A reads the above data. Newly added sensor data won't be read out with 0x3A command.

eWH35_SENSORCH1 = 40,//电压=val*0.02V 当<=1.2V 时显示低电压
eWH35_SENSORCH2 = 41,//电压=val*0.02V 当<=1.2V 时显示低电压

eWH35_SENSORCH3 = 42,//电压=val*0.02V 当<=1.2V 时显示低电压
eWH35_SENSORCH4 = 43,//电压=val*0.02V 当<=1.2V 时显示低电压
eWH35_SENSORCH5 = 44,//电压=val*0.02V 当<=1.2V 时显示低电压
eWH35_SENSORCH6 = 45,//电压=val*0.02V 当<=1.2V 时显示低电压
eWH35_SENSORCH7 = 46,//电压=val*0.02V 当<=1.2V 时显示低电压
eWH35_SENSORCH8 = 47,//电压=val*0.02V 当<=1.2V 时显示低电压

//command 0x3c reads all the data from 0x3A, plus new sensor wh35 can be read. So 0x3C command should be used
//instead of 0x3A if needs to read more sensors to be added.

//fixed sensor data sequence, not allowed for any order change.
eWH90_SENSOR = 48, ,// 0.02V * val(received val)

//----------
eMAX_SENSOR

}SENSOR_IDT;

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_SENSOR_ID	1	0x3A
Size	1	
Checksum	1	checksum

Console return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_SENSOR_ID	1	0x3A
Size	1	
WH65_SENSOR	1	0x01
WH65_ID	4	unsigned long
Battery	1	
Wh65_signal	1	0~4
WH68_SENSOR	1	0x02
WH68_ID	4	unsigned long
battery	1	
WH68_signal	1	0~4
...SENSOR	1	..
..._ID	4	...
battery		
..._signal	1	0~4
Checksum	1	

32) Read Sensors Status

Description	Length	Notes

Fixed header	2	0xffff
CMD_READ_SENSOR_ID_NEW	1	0x3C
Size	1	
Checksum	1	checksum

Console return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_SENSOR_ID_NEW	1	0x3C
Size	2	
WH65_SENSOR	1	0x01
WH65_ID	4	unsigned long
battery	1	
Wh65_signal	1	0~4
WH68_SENSOR	1	0x02
WH68_ID	4	unsigned long
battery	1	
WH68_signal	1	0~4
...SENSOR	1	..
..._ID	4	...
battery		
..._signal	1	0~4
Checksum	1	

33) Write back Sensors ID:

Description	Length	Notes
Fixed header	2	固定 0xffff
CMD_WRITE_SENSOR_ID:	1	0x3B
Size	1	包长度
WH65_SENSOR	1	0x01
WH65_ID	4	Unsigned long
WH68_SENSOR	1	0x02
WH68_ID	4	Unsigned long
...SENSOR	1	SENSOR_IDT
..._ID	4	Unsigned long
Checksum	1	

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_SENSOR_ID	1	0x3B
Size	1	

Resulte	1	0x00:success , 0x01: fail
Checksum	1	checksum

Note：if written ID = 0xFFFFFFFF, tell console to re-learn sensor by force. If written ID = 0xFFFFFFFE, tell console to disable this sensor by force.

34) Read firmware version info:

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_FIRMWARE_VERSION	1	0x50
Size	1	
Checksum	1	

Console WIFI return：

Description	Length	Notes
Fixed header	2	0xffff
CMD_READ_FIRMWARE_VERSION	1	0x50
Size	1	包长度
Versoin length	1	Max value 23Bytes
Version buffer		For example: "EasyWeatherV1.2.0"
Checksum	1	checksum

35) Firmware upgrade

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_UPDATE	1	0x43
Size	1	
ServerIP	4	0xc0a80063 //"192.168.0.99"
ServerPort	2	1~65535
Checksum	1	

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_ UPDATE	1	0x43
Size	1	
Resulte	1	0x00:sucess，0x01: fail
Checksum	1	checksum

If user choose “Update firmware” ， app side send server IP and port number to the module. Console wifi setup a connection with server:

connect success

sent "user1.bin" or "user2.bin"

sent firmware data size (eg: 33334566 bytes)

sent "start"

sent packet(1) //(packet size 1460bytes)

sent "continue"

.....

.....

sent packet(n)

sent "continue"

sent packet (n+1)

sent "end"

1.client connected with server, and send filename, server respond with file length；

2.client have file length received, send server with start， server return with 1st packet of firmware data

3.client have 1st packet firmware data received, send server with continue，server reply with 2nd packet firmware data

……

4.client have (n-1)th firmware data packet received, send server with continue，server reply with nth packet firmware data

5.client have nth packet firmware data received, if all ok, send server with end.

36) Console reboot:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_REBOOT	1	0x40
Size	1	
Checksum	1	

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_REBOOT	1	0x40
Size	1	
Resulte	1	0x00:sucess，0x01: fail
Checksum	1	checksum

37) Reset to default

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_RESET	1	0x41
Size	1	
Checksum	1	

Console WIFI return:

Description	Length	Notes
Fixed header	2	0xffff
CMD_WRITE_RESET	1	0x41
Size	1	
Resulte	1	0x00:sucess，0x01: fail

Checksum	1	checksum