Spectral Color Illuminance Sensor
USB Communication Protocol REV1.0

This protocol is a USB communication mode that sends inquiries, settings and controls to the device through the PC terminal. The instructions are listed in HEX format.

1、 Online command --00

Data format for PC sending:

Instrument Code	Function code
1 Byte	1 Byte
8C	00

PC side sending:8C 00 // Request online

Data format for Device-side reply:

Instrument Code	Function code	data	
1 Byte	1 Byte	14 Byte	
8C	00	10Byte(mode)	4Byte(serial number)

Instrument side reply:8C 00 4F 48 53 50 2D 33 35 30 49 52 BE A0 33 01 //Online success

8C 00 //Online failed

Analysis Based on Response Information--Data Content:

The first 10 bytes are device models:After conversion is "OHSP-350IR "

The last 4 bytes are device serial numbers:0133A0BE=20160702

2、Setting integral time command -- 01

Data format for PC sending:

Instrument Code	Function code	data			
1 Byte	1 Byte	4 Byte			
8C	01	00(L)	01	02	03(H)

PC side sending:8C 01 34 08 00 00 //**Setting Integral Time**

Content description of sending information data:

Data is 4 byte constant:00000834=2100(us)

Data format for Device-side reply :

Instrument Code	Function code
1 Byte	1 Byte
8C	01

Instrument side Reply: 25 01 // Setup successfully

Empty / others // Setup failed

3、Setting integral mode command-- 02

Data format for PC sending:

Instrument Code	Function Code	date
1 Byte	1 Byte	1 Byte
8C	02	00 lock /01 auto

PC side sending：8C 02 01 // **Setting Integral Mode to auto**

Content description of sending information data:

- **Data is 1 byte : 0x00 is lock, 0x01 is Auto**

Data format for Device-side reply：

Instrument Code	Function Code
1 Byte	1 Byte
8C	02

Instrument side Reply： 8C 02 // Setup successfully
Empty / others // Setup failed

4、Read sampling state command-03

Data format for PC sending：

Instrument Code	Function Code
1 Byte	1 Byte
8C	03

PC side sending： 8C 03 // **Read Sampling State**

Data format for Device-side reply：

Instrument Code	Function Code	date				
1 Byte	1 Byte	7 Byte				
8C	03	1Byte(test data state)	1Byte (test state)	1Byte (reserve)	1Byte (test mode)	6、7、8 bytes
		00(already red)/01(not read)	00(testing)/01(test end)		00(stop)/01(single)/02 (continous)	reserve

Instrument side Reply：8C 03 00 01 00 01 2D 33 35 // Return to Data

Analysis Based on Response Information--Data Content:

- **First Byte:00 is data already to read by PC, Auto zeroing after PC read**
- **Second Byte:01 Test End**
- **Third Byte:Reserve**
- **Forth Byte:01 is Single test**

Fifth、Sixth and seventh byte: reserve, No practical significance

5、Read sample data command -- 04

Data format for PC sending：

Instrument Code	Function Code
1 Byte	1 Byte
8C	04

PC side sending：8C 04 // **Read Sample Data**

Data format for Device-side reply：

Instrument Code	Functio n Code	date
1 Byte	1 Byte	7388Byte
8C	04	m_SpectADdata[012…]

Instrument side Reply：8C 04 … // Return data

Analysis Based on Response Information--Data Content:

- **…:initial data Total 3694Points One point is double bytes.So Total 7388bytes**
- **Data range:Range of values per point 0--65535**

6、Reading Integral Time Command-- 05

Data format for PC sending：

Instrument code	Function code
1 Byte	1 Byte
8C	05

PC side sending：8C 05 // **Reading Integral Time**

Data format for Device-side reply ：

frame	Functio n code	data				
1 Byte	1 Byte	5 Byte				1Byte
8C	05	40(L)	42	0F	00(H)	00(Lock)/01(Auto)

Instrument side Reply：8C 05 40 42 0F 00 01 // Return data

Analysis Based on Response Information--Data Content:

- **The first five bytes:Integral Time-big-endian is 000F4240 =1000000 (us) = 1 (S)**
- **The latter byte:Integral Time-01 Auto**
- **Third byte:Reserve**
- **Forth byte :01 is single test**

7、Write Wavelength Command-- 0x0A

Data format for PC sending：

Instrument code	Function code	data

```
Wavelength position and wavelength parameter structuretypedef struct WaveData
{
  uint16_t NULL_V;
  uint16_t Reserve;  // reserve
  uint32_t Reserve;  // reserve
  uint16_t  StartPlace;     //Initial wavelength position 380
  uint16_t MiddlePlace0; //460
  uint16_t MiddlePlace1; //540
  uint16_t MiddlePlace2; //620
  uint16_t MiddlePlace3; //700
  uint16_t EndPlace;        //780
  float fWaveStart;                 //Initial wavelength
  float MiddlefWave0;  //
  float MiddlefWave1;  //
  float MiddlefWave2;  //
  float MiddlefWave3;  //
  float fWaveEnd;    //
  float k0,k1,k2,k3;              //wave date
}GP_WaveData;
GP_WaveData m_WaveData;
```

1 Byte	1 Byte	sizeof(m_WaveData) Byte
8C	0A	sizeof(m_WaveData)

PC side sending：8C 0A … FCS // Setting wave

Content description of sending information data:

- **… : is** m_WaveData structure，such as Right List

Data format for Device-side reply ：

Instrument code	Function code
1 Byte	1 Byte
8C	0A

Instrument side Reply：8C 0A // Setup success
Empty / others // Setup failed

8、Read Wavelength Command-- 0x 0B

Data format for PC sending：

frame	Function code	checksum
1 Byte	1 Byte	1 Byte
8C	0B	Sum

PC side sending：8C 0B // Read setting wave

Data format for Device-side reply ：

frame	Function code	data
1 Byte	1 Byte	sizeof(m_WaveData)Byte
8C	0B	sizeof(m_WaveData)

Instrument side Reply：8C 0B…// return to data

Analysis Based on Response Information--Data Content:

- **… :is** m_WaveDatastructure，such as Right List。

9、Reading Spectral Calibration Data Command-- 0x 0C

Data format for PC sending：

Instrument code	Function code
1 Byte	1 Byte
8C	0C

PC side sending：8C 0C // read spectrum

Data format for Device-side reply ：

Instrument	Functio	data

```
Spectral Calibration structure Explain
typedef struct SpecCalData
{
   uint16_t          NULL_V;
   uint16_t          TC;          // Standard color
temperature value
   float  fLx;                    // Standard lux value
   float  stIntegTime;// integral time unitus,>=10
   float  SpecKs[SPEC1NMDOT];          //
380-1050nm Spectral Calibration Coefficient
}GP_SpecCalData_t;
GP_SpecCalData_t  m_SpecCalData;
```

code	n code	
1 Byte	1 Byte	sizeof(m_SpecCalData)Byte
8C	0C	sizeof(m_SpecCalData)

Instrument side Reply：8C 0C…// return to data

Analysis Based on Response Information--Data Content:

- **… :is**m_SpecCalDatastructure，such as Right List。

10、 Writing Spectral Calibration Data Command-- 0D

Data format for PC sending：

Instrument code	Function code	data
1 Byte	1 Byte	sizeof(m_SpecCalData) Byte
8C	0D	sizeof(m_SpecCalData)

PC side sending：8C 0D …// write spectrum data

Content description of sending information data:

- **… :is**m_SpecCalDatastructure，such as Right List。

Data format for Device-side reply ：

Instrument code	Function code
1 Byte	1 Byte
8C	0D

Instrument side Reply：8C 0D // Setup successfully
Empty / others // Setup failed

11、 Start Sampling Command-- 0x0E

Data format for PC sending：

Instrument code	Function code	command
1 Byte	1 Byte	1 Byte
8C	0E	00stop /01 single/02 continuity

PC side sending：8C 0E 01 // start single test
8C 0E 02 // start continous test

Send Information - Command Description:

- **00 :is stop test**
- **01 :is single test**
- **02:is continous test**

Data format for Device-side reply ：

Instrument code	Function code
1 Byte	1 Byte
8C	0E

Instrument side Reply：8C 0E // Setup successfully

Empty / others // Setup failed

12、 Stop Sampling Command-- 0x25

Data format for PC sending:

Instrument code	Function code
1 Byte	1 Byte
8C	25

PC side sending: 8C 25 // stop test

Send information: 8C 25 equal to 8C 0E 00;

Data format for Device-side reply :

Instrument code	Function code
1 Byte	1 Byte
8C	25

Instrument side Reply: 8C 25 // Setup successfully

Empty / others // Setup failed

13、 Read Test Result Data Command-- 0x13

Data format for PC sending:

Instrument code	Function code
1 Byte	1 Byte
8C	13

PC side sending: 8C 13 // **Read Test Result spectrum** data

Data format for Device-side reply :

frame	Function code	data
1 Byte	1 Byte	sizeof(SpecDataInfo_t)Byte
8C	13	sizeof(SpecDataInfo_t)

Instrument side Reply: 8C 13 … // return to data

Analysis Based on Response Information--Data Content:

- **… :is**SpecDataInfo_tstructure，such as Right List。

```
Test Result spectrum Data structure
typedef struct SpecDataInfo
{
  char  iType[10];  // type
  uint16_t          iVer;   // version 1.00.00
  char  Mark[24];  // description
  float  fLx;     // illuminationE(lx)
  float  fEfc;    // candle lightE(fc)
  float  cct;    // CCT(K)
  float  duv;    // Blackbody deviation Duv
  float  x,y;    // coordinate x,y
  float  u,v;    // coordinate u,v
  float  u2,v2;   // coordinate u',v'
  float  fSDCM;       // Color tolerance SDCM
  float  Ra;     // CRI Ra
  float  Ri[15];   // R1-R15
  float  F;                 // irradiance Ee(W/m2)
  float  fSP;    // S/P
  float  DominantWave;  // Dominant wa(nm)
  float  Purity;        // (%)
  float  HlafWidth; // (nm)
  float  PeakWave; // (nm)
  float  CentreWave;          // (nm)
  float  CentroidWave; // (nm)
  float  fRraito;       // (%)
  float  fGraito;       // (%)
  float  fBraito;       // (%)
  float  X,Y,Z;   // X,Y,Z
  float  IntegTime0; // us
  float  VPeak;       //
  float  VDark;  //
  float  VDarkDAC;  //
  char  TestDate[11];  // 2016-10-11
  char  TestTime[9];  // 12:15:01
  float Spect1nmData[671]; // 380-1050nm
  float  StartTestWave; //
  float  EndTestWave;  //
} GP_SpecDataInfo_t;
GP_SpecDataInfo_t SpecDataInfo_t;
```

14、 Read Test Result Correction Data Command-- 0x26

Data format for PC sending:

Instrument code	Function code
1 Byte	1 Byte

8C	26

PC side sending：8C 26 // Read Test Result Correction data

Data format for Device-side reply ：

Instrument code	Function code	data
1 Byte	1 Byte	sizeof(m_ResultCorrect)Byte
8C	26	sizeof(m_ResultCorrect)

Instrument side Reply：8C 26 … // return to data

Analysis Based on Response Information--Data Content:

- **… :is**m_ResultCorrectstructure，such as Right List。

```
Test Result Correction
typedef struct ResultCorrect
{
   uint16_t  NULL_V;
   uint16_t  NULL_V2;
   float  fStLx;                  //standard value
   float  fLx;                    // Current value
   float  fLxKs;                  //correction factor
   float  Tc[6];                  //
   float  dx[5];                  //
   float  dy[5];                  //
   float  Dw[6];                  //
   float  dDw[5];     //
   float  dRi[15];    //
}GP_ResultCorrect;

GP_ResultCorrect  m_ResultCorrect;
```

15、 Write Test Result Correction Data Command-- 0x27

Data format for PC sending：

Instrument code	Function code	data
1 Byte	1 Byte	sizeof(m_ResultCorrect) Byte
8C	27	sizeof(m_ResultCorrect)

PC side sending：8C 27 … //**Write Test Result Correction Data**

Content description of sending information data:

- **… :is**m_SpecCalDatastructure，such as Right List。

Data format for Device-side reply ：

Instrument code	Function code
1 Byte	1 Byte
8C	27

Instrument side Reply：8C 27 // Setup successfully
Empty / others // Setup failed

```
Test setting structure
typedef struct GP_MeasSetting
{
   uint8_t  NULL_V;
   uint8_t  SaveMode;//auto/manual
   uint16_t  SaveTime;//
   uint8_t  SaveDataFormat;//
   uint8_t  SaveDataFormat_Ohs;//
   uint8_t  SaveDataFormat_Csv;//
   uint8_t  SaveDataStyle;//
   uint16_t          AutoIntegraLimit;//
   uint8_t  AverageTime;//
   uint8_t  ContrastTestMode;//
   uint8_t  MeasFrequency;//
   uint16_t  StartTestWave;//
   uint16_t  EndTestWave;//
} GP_MeasSet_t;
GP_MeasSet_t  Set_MeasSet_t
```

16、 Read Test Setup Data Command-- 0x2A

Data format for PC sending：

Instrument code	Function code
1 Byte	1 Byte
8C	2A

PC side sending：8C 2A //Read Test Setup Data

Data format for Device-side reply ：

Instrument code	Functio n code	data
1 Byte	1 Byte	sizeof(Set_MeasSet_t)Byte
8C	2A	sizeof(Set_MeasSet_t)

Instrument side Reply：8C 2A … //return to data

Analysis Based on Response Information--Data Content:

- **… :is**Set_MeasSet_tstructure，such as Right List。

17、 Write test setup data commands-- 0x2B

Data format for PC sending：

Instrument code	Function code	data
1 Byte	1 Byte	sizeof(Set_MeasSet_t) Byte
8C	2B	sizeof(Set_MeasSet_t)

PC side sending：8C 2B … //**Write test setup data**

Content description of sending information data:

- **… :is**Set_MeasSet_tstructure，such as Right List。

Data format for Device-side reply ：

Instrument code	Function code
1 Byte	1 Byte
8C	2B

Instrument side Reply：8C 2B // Setup successfully

Empty / others // Setup failed

18、 Read System Time Data Command-- 0x2C

Data format for PC sending:

Instrument code	Function code
1 Byte	1 Byte
8C	2C

PC side sending:8C 2C //**Read System Time Data**

Data format for Device-side reply：

frame	Function code	data
1 Byte	1 Byte	16Byte
8C	2C	年(HL)月(HL)日(HL)时(HL)分(HL)秒(HL)保留(HL)

Instrument side Reply:8C 2C E1 07 0C 00 1C 00 0A 00 01 00 22 00 00 00 // return to data

Analysis Based on Response Information--Data Content:

- **First and second bytes: year-little-endian** 07E1=**2017(年)**
- **Third and Four Bytes: Month-little-endian** 000C=**12(月)**
- **Fifth and sixth bytes: day-little-endian** 001C=**28(日)**
- **Seventh and eighth bytes:time-little-endian** 000A=**10(时)**
- **Ninth and tenth bytes:minutes-little-endian** 0001=**01(分)**
- **Eleventh and twelve bytes: seconds-little-endian** 0022=**34(秒)**
- **Thirteenth and fourteenth bytes: reserved**

19、 Write System Time Data Command— 2D

Data format for PC sending:

frame	Function code	data
1 Byte	1 Byte	16Byte
8C	2D	年(HL)月(HL)日(HL)时(HL)分(HL)秒(HL)保留(HL)

PC side sending:8C 2D E1 07 0C 00 1C 00 0A 00 01 00 22 00 00 00//**Write System Time**

Content description of sending information data:

- **First and second bytes: year-little-endian** 07E1=**2017(年)**
- **Third and Four Bytes: Month-little-endian** 000C=**12(月)**
- **Fifth and sixth bytes: day-little-endian** 001C=**28(日)**
- **Seventh and eighth bytes:time-little-endian** 000A=**10(时)**
- **Ninth and tenth bytes:minutes-little-endian** 0001=**01(分)**
- **Eleventh and twelve bytes: seconds-little-endian** 0022=**34(秒)**
- **Thirteenth and fourteenth bytes: reserved**

Data format for Device-side reply：

Instrument code	Function code
1 Byte	1 Byte
8C	2D

Instrument side Reply：8C 2D // Setup successfully
Empty / others // Setup failed

20、 Read Battery Information Command— 0xC3

Data format for PC sending：

Instrument code	Function code
1 Byte	1 Byte
8C	C3

PC side sending： 8C C3 // **Read Battery Information**

Data format for Device-side reply ：

Instrument code	Function code	data		
1 Byte	1 Byte	5Byte		
8C	C3	voltage(HL)	current(HL)Signed number	Residual percentage(0-100%)

Instrument side Reply：8C C3 10 78 FF 39 64 //**return to Battery Information**

Analysis Based on Response Information--Data Content:

- **First and second bytes:Voltage-big-endian** 1078**=4216(mV)**
- **Third and Four Bytes:Current-big-endian FF39=-199(mA)**
- **Fifth bytes:Battery Percentage-** 64**=100(%)**

21、 Read parameter commands for automatic shutdown— C4

Data format for PC sending：

Instrument code	Function code
1 Byte	1 Byte
8C	C4

PC side sending： 8C C4 // **Read parameter commands for automatic shutdown**

Data format for Device-side reply ：

frame	Function code	data	
1 Byte	1 Byte	5Byte	
8C	C4	Atuo shutdown (0 forbid 1 allow)	Shutdown time limit (uint32type)

Instrument side Reply：8C C4 00 00 00 02 58 //**return to automatic shutdown**

Analysis Based on Response Information--Data Content:

- **First byte:state -** 00**=forbid auto shutdown**
- **Second, third, fourth, and fifth bytes:time limit-big-endian is 00000258=600 (S)**

22、 Setting parameter commands for automatic shutdown— 0xC5

Data format for PC sending:

frame	Function code	data	
1 Byte	1 Byte	5Byte	
8C	C5	Atuo shutdown (0 forbid 1 allow)	Shutdown time limit (uint32type)

PC side sending:8C C5 00 00 00 02 58 // 发送设置参数

Content description of sending information data:

- **First byte:state-is** 00**=forbid auto shutdown**
- **Second, third, fourth, and fifth bytes:time limit-big-endian is 00000258=600 (S)**

Data format for Device-side reply :

Instrument code	Function code
1 Byte	1 Byte
8C	C5

Instrument side Reply:8C C5 // Setup successfully
Empty / others // Setup failed

23、 Enter Spectral Calibration Command— 0x19

Data format for PC sending:

Instrument code	Function code
1 Byte	1 Byte
8C	19

PC side sending:8C 19 // **Enter Spectral Calibration**

Data format for Device-side reply :

Instrument code	Function code
1 Byte	1 Byte
8C	19

Instrument side Reply:8C 19 // enter success
Empty / others // enter failed

24、 Enter Spectral Calibration KEY Command — 0x1A

Data format for PC sending:

Instrument code	Function code	data
1 Byte	1 Byte	1Byte
8C	1A	01 cancel 02 stop 03 save

PC side sending: 8C 1A 01 // **Exit Spectral Calibration**
PC side sending: 8C 1A 02 // **Calibration/Stop Spectral Calibration**
PC side sending: 8C 1A 03 // **save data**

Data format for Device-side reply：

frame	Data length		Function code	data	checksum
1 Byte	2 Byte		1 Byte	1 Byte	1 Byte
8C	00(H)	01(L)	1A	00 failed /01 success	Sum

Instrument side Reply: 8C 1A // Setup successfully
Empty / others // Setup failed