



TB200B 智能气体传感器模组

AQS用户通讯协议

UART接口

本文件详细介绍了TB200B智能气体传感器模组的AQS通信模式，用户在使用产品时所需要使用的主要指令。

1. 产品概述和配置参数

1.1 产品概述

该产品使用UART进行通信，具体配置请参阅【1.2 UART配置参数】。

该产品具有两种传输方式：查询模式和主动上报模式，具体请查阅【1.3 通讯模式】。

1.2 UART配置参数

| 波特率 | 数据位 | 起始位 | 停止位 | 校验位 | 读写数据间隔 |
|------|-----|-----|-----|-----|--------|
| 9600 | 8位 | 1位 | 1位 | 无 | >=1s |

1.3 通讯模式

模式一：查询模式

此模式下，上位机发送指令进行查询指定数据。此模式为默认模式，在每次上电或从休眠中唤醒后，都将进入此模式。

模式二：自动上报模式

此模式下，每隔1s会自动上发一组数据，这组数据包括气体浓度值和量程。

两种模式可以使用指令进行切换，具体指令请参照【4.1 模式切换指令】

2. 气体浓度单位表

| 值 (Hex) | 0x02 | 0x04 | 0x08 |
|----------------------|-------------------|-------------------|--------------------|
| 对应Concentration-1的单位 | ppm | ppb | %vol. |
| 对应Concentration-2的单位 | mg/m ³ | ug/m ³ | 10g/m ³ |

模组在生产时采用的Concentration-1的单位进行标定，Concentration-2通过浓度转换公式计算得出。

并且量程也是采用Concentration-1的单位。

Concentration-1和Concentration-2的浓度值转换公式为： $\text{mg/m}^3 = \text{M}/22.4 \times \text{ppm} \times [273/(273+T)] \times (\text{Ba}/101325)$

M：分子量

T：温度（摄氏度）

Ba：气压

在模组内部转换中Ba为1个标准大气压，T为25摄氏度。

其中VOC是未知分子量，所以Concentration-2和Concentration-1的转换比为1：1

3. 传感器类型表

| 传感器类型 | HCHO | VOC | CO | Cl ₂ | H ₂ | H ₂ S | HCl | HCN | HF | NH ₃ | NO ₂ | O ₂ | O ₃ | SO ₂ |
|---------|------|------|------|-----------------|----------------|------------------|------|------|------|-----------------|-----------------|----------------|----------------|-----------------|
| 值 (Hex) | 0x17 | 0x18 | 0x19 | 0x1A | 0x1B | 0x1C | 0x1D | 0x1E | 0x1F | 0x20 | 0x21 | 0x22 | 0x23 | 0x24 |

| | | | | | | | | | | | | | | |
|---------|---------------------------------|---------------------------------|----------------------------------------------|---------------------------------|---------------------------------|-----------------------------------------------|-------------------------------|----------------------------------------------|-------------------------------|--------------------------------|--------------------------------|-------------------------------|--------------------------------|---------------------------------|
| 传感器类型 | HBr | Br ₂ | F ₂ | PH ₃ | AsH ₃ | SiH ₄ | GeH ₄ | B ₂ H ₆ | BF ₃ | WF ₆ | SiF ₄ | XeF ₂ | TiF ₄ | SMELL |
| 值 (Hex) | 0x25 | 0x26 | 0x27 | 0x28 | 0x29 | 0x2A | 0x2B | 0x2C | 0x2D | 0x2E | 0x2F | 0x30 | 0x31 | 0x32 |
| 传感器类型 | IAQ | AQI | NMHC | SO _x | NO _x | NO | C ₄ H ₈ | C ₃ H ₈ O ₂ | CH ₄ S | C ₈ H ₈ | C ₄ H ₁₀ | C ₂ H ₆ | C ₆ H ₁₄ | C ₂ H ₄ O |
| 值 (Hex) | 0x33 | 0x34 | 0x35 | 0x36 | 0x37 | 0x38 | 0x39 | 0x3A | 0x3B | 0x3C | 0x3D | 0x3E | 0x3F | 0x40 |
| 传感器类型 | C ₃ H ₉ N | C ₂ H ₇ N | C ₂ H ₆ O | CS ₂ | C ₂ H ₆ S | C ₂ H ₆ S ₂ | C ₂ H ₄ | CH ₃ OH | C ₆ H ₆ | C ₈ H ₁₀ | C ₇ H ₈ | CH ₃ COOH | ClO ₂ | |
| 值 (Hex) | 0x41 | 0x42 | 0x43 | 0x44 | 0x45 | 0x46 | 0x47 | 0x48 | 0x49 | 0x4A | 0x4B | 0x4C | 0x4D | |
| 传感器类型 | H ₂ O ₂ | N ₂ H ₄ | C ₂ H ₈ N ₂ | C ₂ HCl ₃ | CHCl ₃ | C ₂ H ₃ Cl ₃ | H ₂ Se | | | | | | | |
| 值 (Hex) | 0x4E | 0x4F | 0x50 | 0x51 | 0x52 | 0x53 | 0x54 | | | | | | | |

4. 具体交互指令和例程

4.1 通讯模式切换指令

- 切换到自动上报模式

下行指令格式：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0xFF | 0x01 | 0x78 | 0x40 | 0x00 | 0x00 | 0x00 | 0x00 | 0x47 |

上行数据格式 (每隔1s)：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|--------------------------|---------|----------------|---------|--------------------------|---------|----------|
| 0xFF | 0x86 | Concentration-2 (uint16) | | Range (uint16) | | Concentration-1 (uint16) | | Checksum |

切换到自动上报模式示例：

发送: 0xFF 0x01 0x78 0x40 0x00 0x00 0x00 0x00 0x47

接收: 0xFF 0x86 0x25 0xBC 0x03 0xE8 0x20 0xD0 0xBE

发送解析：

发送数据为固定格式

接收解析：

0xFF 0x86-----指令头

0x25 0xBC-----0x25BC=9660 Concentration-2根据【4.2 获取模组参数】章节中的参数计算得到9.66 mg/m³

0x03 0xE8-----0x03E8=1000 Range根据【4.2 获取模组参数】章节中的参数计算得到1000 ppm

0x20 0xD0-----0x20D0=8400 Concentration-1根据【4.2 获取模组参数】章节中的参数计算得到8.4 ppm

0xBE-----校验和 (-(0x86+0x25+0xBC+0x03+0xE8+0x20+0xD0))&0xFF

• 切换到查询模式

下行指令格式：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0xFF | 0x01 | 0x78 | 0x41 | 0x00 | 0x00 | 0x00 | 0x00 | 0x46 |

切换到查询模式示例：

发送: 0xFF 0x01 0x78 0x41 0x00 0x00 0x00 0x00 0x46

4.2 查询模组参数 (方式一)

下行指令格式：

| Byte[0] |
|---------|
| 0xD1 |

上行数据格式：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|-------------|----------------|---------|-----------|----------|---------|---------|---------------|----------|
| Sensor type | Range (uint16) | | Unit type | Reserved | | | Digital place | Checksum |

查询模组参数 (方式一) 示例：

发送: 0xD1

接收: 0x19 0x03 0xE8 0x02 0x00 0x00 0x00 0x30 0xE3

发送解析：

发送数据为固定格式

接收解析：

0x19-----传感器类型，请参阅【3. 传感器类型表】
0x03 0xE8-----0x03E8=1000 Range根据【4.2 获取模组参数】章节中的参数计算得到1000 ppm
0x02-----Concentration-2单位为mg/m³，Concentration-1的单位为ppm，具体请参阅【2. 气体浓度单位表】
0x00 0x00 0x00-----保留
0x30-----小数位数，0x30>>4=0x03，则在计算Concentration-2和Concentration-1时需要将原始数据除以1000
0xE3-----校验和 (-(0x03+0xE8+0x02+0x00+0x00+0x00+0x30))&0xFF

4.3 查询模组参数 (方式二)

此方式和【4.2 查询模组参数 (方式一)】的功能相同，指令格式不同，用户可以自行选择使用。

下行指令格式：

| Byte[0] |
|---------|
| 0xD7 |

上行数据格式：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|-------------|----------------|---------|-----------|---------------|----------|----------|
| 0xFF | 0xD7 | Sensor type | Range (uint16) | | Unit type | Digital place | Reserved | Checksum |

查询模组参数 (方式二) 示例:

发送: 0xD7

接收: 0xFF 0xD7 0x19 0x03 0xE8 0x02 0x30 0x00 0xF3

发送解析:

发送数据为固定格式

接收解析:

0xFF 0xD7-----指令头
 0x19-----传感器类型, 请参阅【3. 传感器类型表】
 0x03 0xE8-----0x03E8=1000 Range根据【4.2 获取模组参数】章节中的参数计算得到1000 ppm
 0x02-----Concentration-2单位为mg/m³ Concentration-1的单位为ppm, 具体请参阅【2. 气体浓度单位表】
 0x30-----小数位数, 0x30>4=0x03, 则在计算Concentration-2和Concentration-1时需要将原始数据除以1000
 0xE3-----校验和 $(-(0xD7+0x19+0x03+0xE8+0x02+0x30+0x00))\&0xFF$

4.4 查询传感器当前浓度值

下行指令格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0xFF | 0x01 | 0x86 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x79 |

上行数据格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|--------------------------|---------|----------------|---------|--------------------------|---------|----------|
| 0xFF | 0x86 | Concentration-2 (uint16) | | Range (uint16) | | Concentration-1 (uint16) | | Checksum |

查询传感器当前浓度值示例:

发送: 0xFF 0x01 0x86 0x00 0x00 0x00 0x00 0x00 0x79

接收: 0xFF 0x86 0x25 0xBC 0x03 0xE8 0x20 0xD0 0xBE

发送解析:

发送数据为固定格式

接收解析:

0xFF 0x86-----固定头
 0x25 0xBC-----0x25BC=9660 Concentration-2根据【4.2 获取模组参数】章节中的参数计算得到9.66 mg/m³
 0x03 0xE8-----0x03E8=1000 Range根据【4.2 获取模组参数】章节中的参数计算得到1000 ppm
 0x20 0xD0-----0x20D0=8400 Concentration-1根据【4.2 获取模组参数】章节中的参数计算得到8.4 ppm
 0xBE-----校验和 $(-(0x86+0x25+0xBC+0x03+0xE8+0x20+0xD0))\&0xFF$

4.5 查询传感器当前浓度值以及温湿度

下行指令格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0xFF | 0x01 | 0x87 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x78 |

上行数据格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] | Byte[9] | Byte[10] | Byte[11] | Byte[12] |
|---------|---------|--------------------------|---------|----------------|---------|--------------------------|---------|---------------------|---------|-------------|----------|----------|
| 0xFF | 0x87 | Concentration-2 (uint16) | | Range (uint16) | | Concentration-1 (uint16) | | Temperature (int16) | | Rh (uint16) | | Checksum |

查询传感器当前浓度值以及温湿度示例:

发送: 0xFF 0x01 0x87 0x00 0x00 0x00 0x00 0x00 0x78

接收: 0xFF 0x87 0x25 0xBC 0x03 0xE8 0x20 0xD0 0x07 0x3B 0x21 0x07 0x53

发送解析:

发送数据为固定格式

接收解析:

0xFF 0x87-----固定头

0x25 0xBC-----0x25BC=9660 Concentration-2根据【4.2 获取模组参数】章节中的参数计算得到9.66 mg/m³

0x03 0xE8-----0x03E8=1000 Range根据【4.2 获取模组参数】章节中的参数计算得到1000 ppm

0x20 0xD0-----0x20D0=8400 Concentration-1根据【4.2 获取模组参数】章节中的参数计算得到8.4 ppm

0x07 0x3B-----0x073B=1851 Temperature 固定2位小数, 18.51摄氏度

0x21 0x07-----0x2107=8455 Rh 固定两位小数, 84.55% Rh

0x53-----校验和 (-(0x87+0x25+0xBC+0x03+0xE8+0x20+0xD0+0x07+0x3B+0x21+0x07))&0xFF

4.6 查询温度和相对湿度 (方式一)

下行指令格式:

| Byte[0] |
|---------|
| 0xD2 |

上行数据格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] |
|---------------------|---------|-------------|---------|
| Temperature (int16) | | Rh (uint16) | |

查询温度和相对湿度示例:

发送: 0xD2

接收: 0x07 0x3B 0x21 0x07

发送解析:

发送数据为固定格式

接收解析:

0x07 0x3B-----0x073B=1851 Temperature 固定2位小数, 18.51摄氏度

0x21 0x07-----0x2107=8455 Rh 固定两位小数, 84.55% Rh

4.7 查询温度和相对湿度 (方式二)

此方式和【4.6 查询温度和相对湿度(方式一)】的功能相同, 指令格式不同, 用户可以自行选择使用。

下行指令格式:

| Byte[0] |
|---------|
| 0xD6 |

上行数据格式：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] |
|---------------------|---------|-------------|---------|----------|
| Temperature (int16) | | Rh (uint16) | | Checksum |

查询温度和相对湿度示例:

发送: 0xD6

接收: 0x07 0x3B 0x21 0x07 0x96

发送解析:

发送数据为固定格式

接收解析:

0x07 0x3B-----0x073B=1851 Temperature 固定2位小数, 18.51摄氏度

0x21 0x07-----0x2107=8455 Rh 固定两位小数, 84.55% Rh

0x96-----校验和 $-(0x07+0x3B+0x21+0x07)\&0xFF$

4.8 查询软件版本号

下行指令格式:

| Byte[0] |
|---------|
| 0xD3 |

上行数据格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] |
|------------------------|---------|---------|---------|---------|---------|
| Software version (BCD) | | | | | |

查询软件版本号示例:

发送: 0xD3

接收: 0x20 0x23 0x11 0x08 0x14 0x54

发送解析:

发送数据为固定格式

接收解析:

0x20 0x23 0x11 0x08 0x14 0x54-----0x202311081454 按BCD码转换后 202311081454

4.9 进入休眠模式 (方式一)

使用此条指令进入休眠后, 当需要退出休眠时, 必须使用【4.10 退出休眠模式 (方式一)】指令进行退出。

下行指令格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] |
|---------|---------|---------|---------|---------|---------|
| 0xAF | 0x53 | 0x6C | 0x65 | 0x65 | 0x70 |

上行数据格式:

| Byte[0] | Byte[1] |
|---------|---------|
| 0x4F | 0x4B |

进入休眠模式（方式一）示例:

发送: 0xAF 0x53 0x6C 0x65 0x65 0x70

接收: 0x4F 0x4B

发送解析:

发送数据为固定格式

接收解析:

接收数据为固定格式

4.10 退出休眠模式（方式一）

使用此条指令于【4.9 进入休眠模式（方式一）】指令搭配使用。退出休眠模式后需要等待6s，才能正常读取数据。

下行指令格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] |
|---------|---------|---------|---------|---------|
| 0xAE | 0x45 | 0x78 | 0x69 | 0x74 |

上行数据格式:

| Byte[0] | Byte[1] |
|---------|---------|
| 0x4F | 0x4B |

退出休眠模式（方式一）示例:

发送: 0xAE 0x45 0x78 0x69 0x74

接收: 0x4F 0x4B

发送解析:

发送数据为固定格式

接收解析:

接收数据为固定格式

4.11 进入休眠模式（方式二）

使用此条指令进入休眠后，当需要退出休眠时，必须使用【4.12 退出休眠模式（方式二）】指令进行退出。

下行指令格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] |
|---------|---------|---------|---------|---------|---------|---------|
| 0xA1 | 0x53 | 0x6C | 0x65 | 0x65 | 0x70 | 0x32 |

上行数据格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0xFF | 0xA1 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x5F |

进入休眠模式 (方式二) 示例:

发送: 0xA1 0x53 0x6C 0x65 0x65 0x70 0x32

接收: 0xFF 0xA1 0x00 0x00 0x00 0x00 0x00 0x00 0x5F

发送解析:

发送数据为固定格式

接收解析:

接收数据为固定格式

4.12 退出休眠模式 (方式二)

使用此条指令于【4.11 进入休眠模式 (方式二)】指令搭配使用。退出休眠模式后需要等待6s, 才能正常读取数据。

下行指令格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] |
|---------|---------|---------|---------|---------|---------|
| 0xA2 | 0x45 | 0x78 | 0x69 | 0x74 | 0x32 |

上行数据格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0xFF | 0xA2 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x5E |

退出休眠模式 (方式二) 示例:

发送: 0xA2 0x45 0x78 0x69 0x74 0x32

接收: 0xFF 0xA2 0x00 0x00 0x00 0x00 0x00 0x00 0x5E

发送解析:

发送数据为固定格式

接收解析:

接收数据为固定格式

4.13 关闭运行灯

下行指令格式:

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0xFF | 0x01 | 0x88 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x77 |

上行数据格式:

| Byte[0] | Byte[1] |
|---------|---------|
| 0x4F | 0x4B |

关闭运行灯示例:

发送: 0xFF 0x01 0x88 0x00 0x00 0x00 0x00 0x00 0x77

接收: 0x4F 0x4B

发送解析:

发送数据为固定格式

接收解析:

接收数据为固定格式

4.14 打开运行灯

下行指令格式：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0xFF | 0x01 | 0x89 | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x76 |

上行数据格式：

| Byte[0] | Byte[1] |
|---------|---------|
| 0x4F | 0x4B |

打开运行灯示例：

发送：0xFF 0x01 0x89 0x00 0x00 0x00 0x00 0x00 0x76

接收：0x4F 0x4B

发送解析：

发送数据为固定格式

接收解析：

接收数据为固定格式

4.15 查询运行灯状态

下行指令格式：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0xFF | 0x01 | 0x8A | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x75 |

上行数据格式：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|--------------|----------|---------|---------|---------|---------|----------|
| 0xFF | 0x8A | Light status | Reserved | | | | | Checksum |

查询运行灯示例：

发送：0xFF 0x01 0x8A 0x00 0x00 0x00 0x00 0x00 0x75

接收：0xFF 0x8A 0x01 0x00 0x00 0x00 0x00 0x00 0x75

发送解析：

发送数据为固定格式

接收解析：

0xFF 0x8A-----指令头

0x01-----0x01：打开 0x00：关闭

0x00 0x00 0x00 0x00 0x00-----保留

0x75-----校验和 $(-(0x8A+0x01+0x00+0x00+0x00+0x00+0x00))\&0xFF$

4.16 用户标定浓度

Float类型数据转换：请参考http://www.binaryconvert.com/convert_float.html?

C语言可参看如下联合体进行转换

```
typedef union {
    float data;
    uint32_t uint32_data;
    uint8_t uint8_data[4];
}FLOAT_DATA_U;
```

下行指令格式：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|----------------------------------------|---------|---------|---------|---------|----------|
| 0xFF | 0x01 | 0x8D | Concentration (float)(\leq range/2) | | | | 0x00 | Checksum |

上行数据格式：

| Byte[0] | Byte[1] |
|---------|---------|
| 0x4F | 0x4B |

用户标定浓度示例：

例如写入10ppm，10.0f 按照IEEE754 single precision 32-bit转换后的十六进制整数为：0x41200000

发送：0xFF 0x01 0x8D 0x41 0x20 0x00 0x00 0x00 0x11

接收：0x4F 0x4B

发送解析：

发送数据为固定格式

接收解析：

0xFF 0x01 0x8D-----指令头
 0x41 0x20 0x00 0x00-----0x41200000 — 10.0f (IEEE754 single precision 32-bit)
 0x00-----保留
 0x11-----校验和 $(-(0x01+0x8D+0x41+0x20+0x00+0x00+0x00))\&0xFF$

4.17 恢复出厂标定

下行指令格式：

| Byte[0] | Byte[1] | Byte[2] | Byte[3] | Byte[4] | Byte[5] | Byte[6] | Byte[7] | Byte[8] |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|
| 0xFF | 0x01 | 0x8E | 0x00 | 0x00 | 0x00 | 0x00 | 0x00 | 0x71 |

上行数据格式：

| Byte[0] | Byte[1] |
|---------|---------|
| 0x4F | 0x4B |

恢复出厂标定示例：

发送：0xFF 0x01 0x8E 0x00 0x00 0x00 0x00 0x00 0x71

接收：0x4F 0x4B

发送解析：

发送数据为固定格式

接收解析：

接收数据为固定格式



德国研发生产中心

德国 EC Sense GmbH

Wangener Weg 3 | 82069 Hohenschäftlarn

座机: +49 (0)8178-99992-10

传真: +49 (0)8178-99992-11

邮箱: office@ecsense.com

网址: www.ecsense.com, www.ecnose.de

亚太区·中国应用设计研发中心

宁波爱氮森科技有限公司

浙江·宁波市鄞州区金谷北路 228 号中物科技园 17 幢 4 层

邮编: 315100

座机: 0574-88097236, 88096372

邮箱: info@aqsystems.cn

网址: www.ecsense.cn, www.ecnose.com