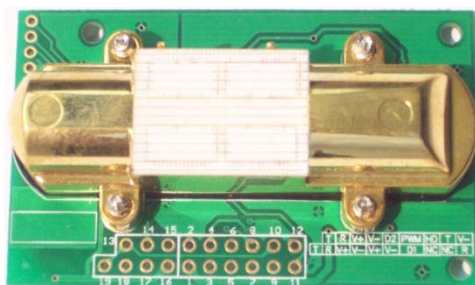


## AT1-CO2 二氧化碳传感器

### 产品描述

AT1-CO2二氧化碳气体传感器是一个智能通用型、小型传感器，利用非色散红外（NDIR）原理对空气中存在的CO<sub>2</sub>进行探测，具有很好的选择性和无氧气依赖性，寿命长。内置温度补偿；同时具有数字输出与模拟电压输出，方便使用。该传感器是将成熟的红外吸收气体检测技术与精密光路设计、精良电路设计紧密结合而制作出的高性能传感器。



### 传感器特点

- 高灵敏度、高分辨率、低功耗
- 提供UART、模拟电压信号、PWM波形等多种输出方式
- 响应时间快
- 温度补偿，卓越的线性输出
- 优异的稳定性
- 使用寿命长
- 抗水汽干扰、不中毒

### 主要应用

可广泛应用于暖通制冷与室内空气质量监控、工业过程及安全防护监控、农业及畜牧业生产过程监控。

### 技术指标

表 1

产品型号	AT1-CO2
检测气体	二氧化碳
工作电压	4.5~5.5V DC
平均电流	< 85 mA
接口电平	3.3V
测量范围	0~5%VOL 范围内可选（详见表 2）
输出信号	0.4~2V DC
	UART
	PWM
预热时间	3min
响应时间	T <sub>90</sub> < 90s
工作温度	0~50℃
工作湿度	0~95%RH（无凝结）
外形尺寸	57.5×34.7×16mm（L×W×H）
重 量	15 g
寿 命	>5 年

量程和精度

表 2

气体名称	分子式	量程	精度	备注
二氧化碳	CO <sub>2</sub>	0~2000ppm	±(50ppm+ 5%读数值)	温度补偿
		0~5000ppm		温度补偿
		0~1%VOL		温度补偿
		0~3%VOL		温度补偿
		0~5%VOL		温度补偿

传感器结构图

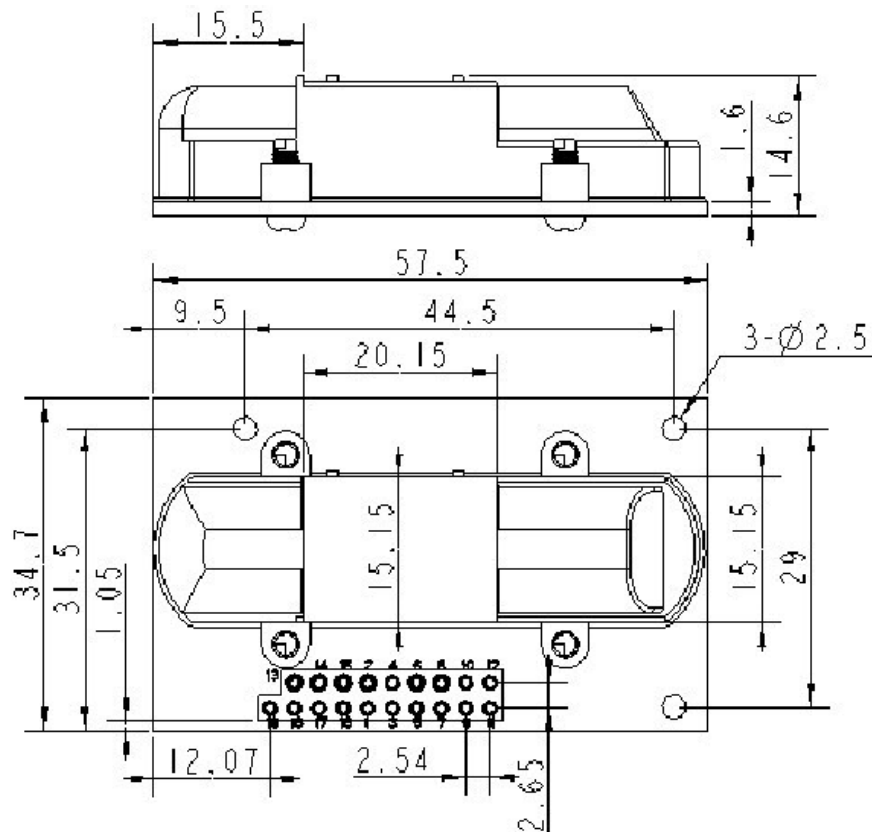


图 1: 传感器结构图

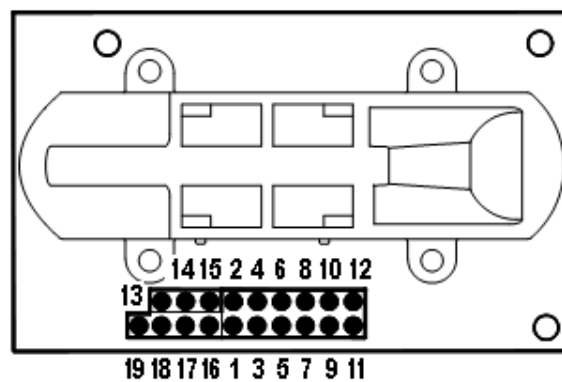


图 2: 管脚定义图

管脚定义

表 3

管脚名称	管脚说明
Pad1、 Pad15、 Pad17	Vin (电压输入 4.5V~5.5V)
Pad2、 Pad3、 Pad12、 Pad16	GND
Pad4	Vout2 (0.4~2V)
Pad5	Vout1 (0~2.5V)
Pad6	PWM
Pad8	HD (校零, 低电平 7 秒以上有效)
Pad7、 Pad9	NC
Pad11、 Pad14、 Pad18	UART (RXD) 0~3.3V 数据输入
Pad10、 Pad13、 Pad19	UART (TXD) 0~3.3V 数据输出

应用电路

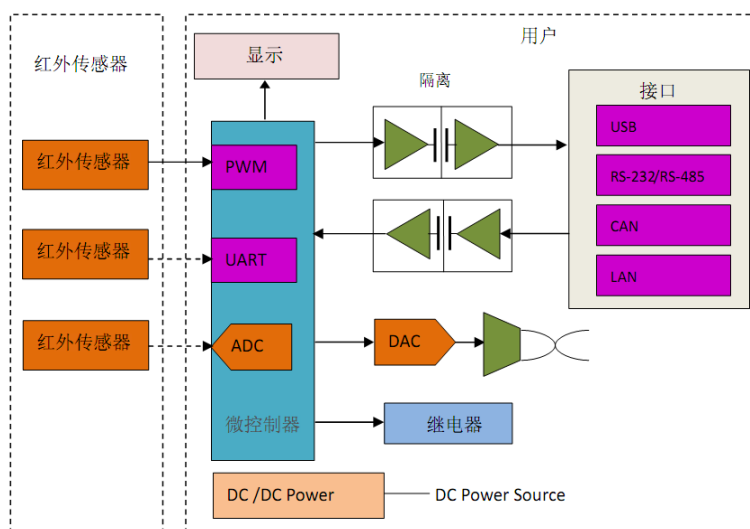


图3：应用电路

说明：1、模拟电压输出

Vout1 输出电压范围可定制，默认 0~2.5V，对应气体浓度（0~满量程）

Vout2 输出电压范围（0.4~2V），对应气体浓度（0~满量程）

将传感器Vin端接5V，GND端接电源地，Vout2端接ADC的输入端。传感器经过预热时间后从Vout2端输出表征气体浓度的电压值，0.4~2.0V 代表气体浓度值0~满量程。当自检发现故障时，传感器输出电压为0V。

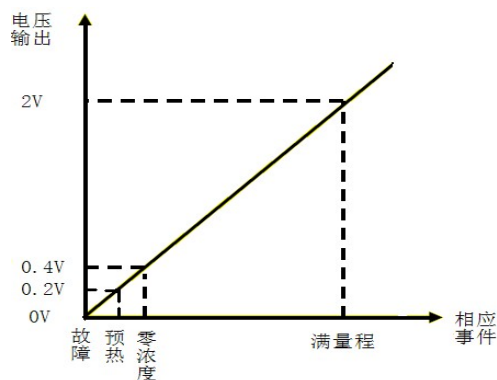


图4：模拟电压输出图示

## 2、PWM 输出

以测量范围为 2000ppm 的 PWM 输出为例：

CO <sub>2</sub> 浓度输出范围	0~2000ppm
周期	1004ms ± 5%
周期起始段高电平输出	2ms (名义上)
中部周期	1000ms ± 5%
周期结束段低电平输出	2ms (名义上)

通过 PWM 获得当前 CO<sub>2</sub> 浓度值的计算公式：

$$C_{ppm} = 2000 \times (T_H - 2ms) / (T_H + T_L - 4ms)$$

其中：

$C_{ppm}$  为通过计算得到的 CO<sub>2</sub> 浓度值，单位为 ppm；

$T_H$  为一个输出周期中输出为高电平的时间；

$T_L$  为一个输出周期中输出为低电平的时间；

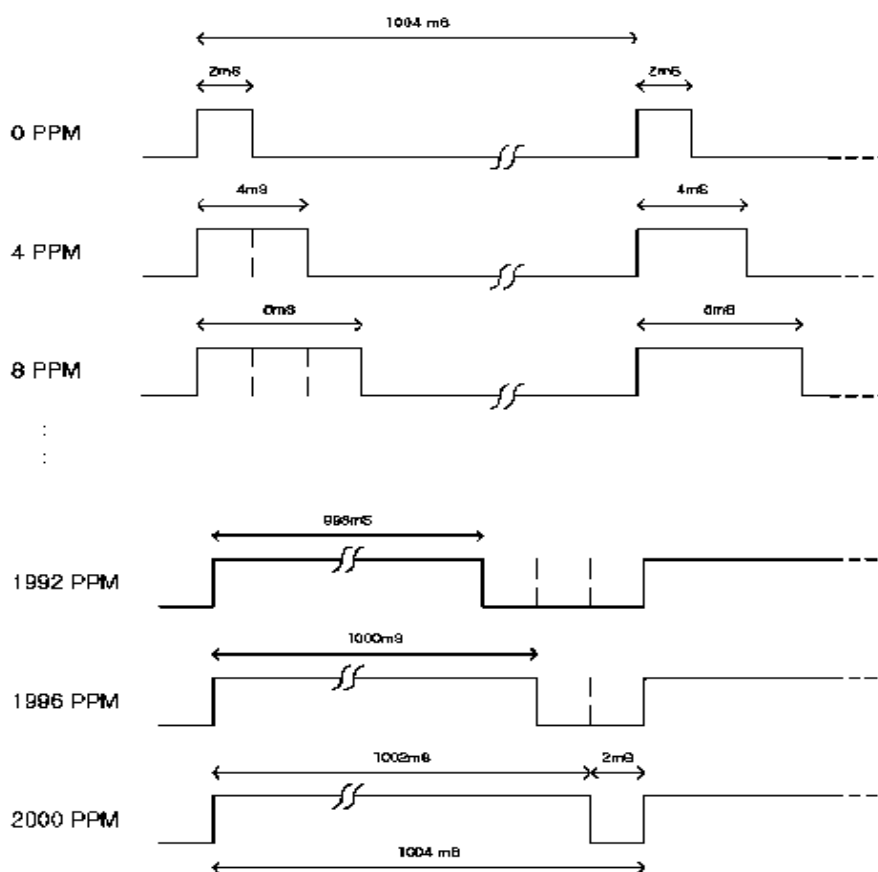


图 5： PWM 输出图示

## 3、数字方式

将传感器 Vin 端接 5V，GND 端接电源地，用户通讯接口的 RXD 端接探测器的 TXD，TXD 端接探测器的 RXD。探测器可以直接通过传感器的 UART 接口读出气体浓度值，不需要计算。

### 3.1 通讯协议

3.1.1 通用设置

表 4

波特率	9600
数据位	8 位
停止位	1 位
校验位	无

3.1.2 命令

每条命令或返回:

包含 9 字节 (字节 0~字节 8)

起始字节固定为 0xff

命令包含传感器编号 (出厂默认值为 0x01)

以校验和结尾 (校验和计算方法见 校验和计算 )

命令列表

表 5

0x86	读气体浓度值
0x87	校准传感器 零点 (ZERO)
0x88	校准传感器 跨度点 (SPAN)

读气体浓度值

发送命令								
Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
起始字节	传感器编号	命令	-	-	-	-	-	校验值
0XFF	0x01	0x86	0x00	0x00	0x00	0x00	0x00	0x79

传感器返回值

传感器返回 (例)								
Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte 8
起始字节	命令	浓度值高位	浓度值低位	-	-	-	-	校验值
0XFF	0x86	0x02	0x60	0x47	0x00	0x00	0x00	0xD1

$$\text{气体浓度值} = \text{浓度值高位} * 256 + \text{浓度值低位}$$

校准传感器零点

发送命令								
Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
起始字节	传感器编号	命令	-	-	-	-	-	校验值
0XFF	0x01	0x87	0x00	0x00	0x00	0x00	0x00	0x78

传感器无返回值

校准传感器跨度值

发送命令								
Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
起始字节	传感器编号	命令	跨度值高位	跨度值低位	-	-	-	校验值
0XFF	0x01	0x88	0x07	0xD0	0x00	0x00	0x00	0xA0

传感器无返回值

### 3.1.3 校验和计算

校验和 = (取反(字节 1+……+字节 7)) +1

例如读气体浓度值:

发送命令								
Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
起始字节	传感器编号	命令	-	-	-	-	-	校验值
0xFF	0x01	0x86	0x00	0x00	0x00	0x00	0x00	0x79

1. 除字节 0 以外的其他字节全部相加

$$0x1 + 0x86 + 0 + 0 + 0 + 0 + 0 = 0x87$$

2. 第一步得到的值取反

$$0xff - 0x87 = 0x78$$

3. 第二步得到的值加 1

$$0x78 + 0x01 = 0x79$$

### 3.2 示例程序

C 语言计算校验和例程

```

char getChecksum(char *packet)
{
    char i, checksum;
    for( i = 1; i < 8; i++)
    {
        checksum += packet[i];
    }
    checksum = 0xff - checksum;
    checksum += 1;
    return checksum;
}
    
```

#### 注意事项

- 1、 传感器应定期标定，建议标定周期 6 个月。
- 2、 不要在粉尘密度大的环境长期使用传感器。
- 3、 请在传感器供电范围内使用传感器。