



深圳市阿尔森物联技术有限公司

Shenzhen arsend IOT Dev.Co.,Ltd

T6713 系列 CO2 传感器应用说明



深圳市阿尔森物联技术有限公司

Shenzhen arsend IOT Dev.Co.,Ltd

修订记录

版次	创作人	发布时间
1	Norman Hannotte	2014-2-19

变更说明

- 首次发布

目录

1	前言	4
2	接口连接器	4
2.1	启动过程中配置 I/O	5
2.1.1	第一种 I/O 配置——“UART”和“SLOW PWM”	5
2.1.2	第二种 I/O 配置——“I2C”和“FAST PWM”	6
2.1.3	第三种 I/O 配置——“UART”和“RS485”	7
3	通信 – MODBUS 协议	7
3.1	UART (RS232/RS485)	7
3.2	I ² C	8
4	指令汇总	8
4.1	FIRMEWARE REVISION 指令	9
4.2	STATUS 指令	11
4.3	GAS PPM 指令	13
4.4	重置设备	16
4.5	START SINGLE POINT CALIBRATION 指令	17
4.6	CHANGE SLAVE ADDRESS 指令	21

1 前言

为明确 T6713 系列 CO2 传感器接口设计和通信协议的相关要求，特编制此文件。本文件的目标受众为希望利用 I²C 或 UART 查询传感器信息的开发人员。

2 接口连接器

印刷电路板上的六针通孔连接器位于传感器的电源及 IO 模块处，参加下图 1。为实现传感器与控制器的连接，必须安装一个 0.1" 的 6 针插头。本系列传感器支持三种不同的 I/O 组态。I/O 具体组态由启动时在 6# 针脚探测到的条件确定。



- 1. Tx/SCL**
- 2. Rx/SDA**
- 3. V++**
- 4. GND**
- 5. PWM**

图 1 - T6713 接口连接器

注：为遵守限制性规定并防止因静电放电或粗暴搬运造成损坏，请采取适当的预防措施。关于预防静电破坏与电子装配正确做法的详细信息，请分别参考 ANSI/ESD S20.20-1999 与 IPC 610 版本 D。此外须注意，本系列传感器不带防反接功能。注意确保以正确的接线形式完成传感器与控制器的连接，以防损坏。

2.1 启动过程中配置 I/O

2.1.1 第一种 I/O 配置——“UART”和“SLOW PWM”

用户须让 6# 针脚保持不连接状态，即让其悬空。通过内部的 1 MΩ 电阻阻断。在此配置下：

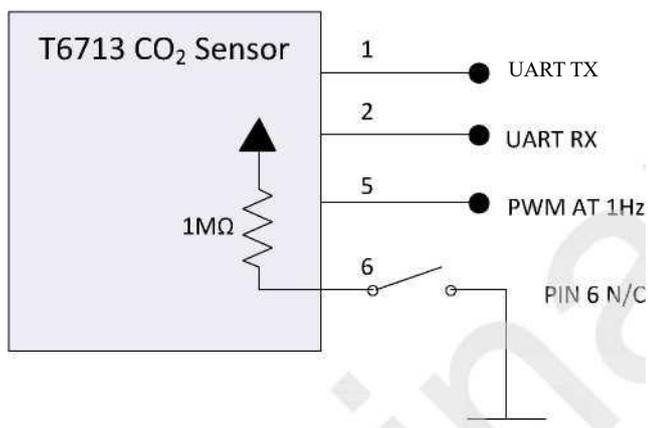


图 2 - 第一种 I/O 配置简图

表 1 - 第一种 I/O 针脚配置

针脚	说明
1	UART TX (传感器输出)
2	UART Rx (传感器输入)
5	PWM 输出 (约 1Hz)

在此配置下，传感器采用 RS-232 串行接口。默认串行通信数据为：

- 19200 波特
- 1 个起始位
- 8 个数据位
- 1 个偶校验位
- 1 个停止位

2.1.2 第二种 I/O 配置——“I2C”和“FAST PWM”

6#针脚由用户接地。在此配置下：

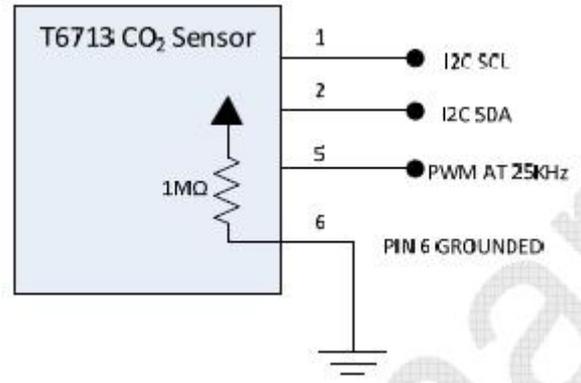


图 3 – 第二种 I/O 配置简图

表 2 – 第二种 I/O 针脚配置

针脚	说明
1	I ² C SCL (串行时钟线路)
2	I ² C SDA (串行日期线路)
5	PWM 输出 (约 25kHz)

在此配置下，传感器采用一个 I²C 接口。默认配置为：

- 传感器仅被用作从设备。
- 采用 I²C 7 位寻址。默认从地址为 21 (0x15)
- **I²C 100kbit/s 标准模式**

2.1.3 第三种 I/O 配置——“UART”和“RS485”

6#针脚通过 10kQ ~100kQ 的电阻接地。

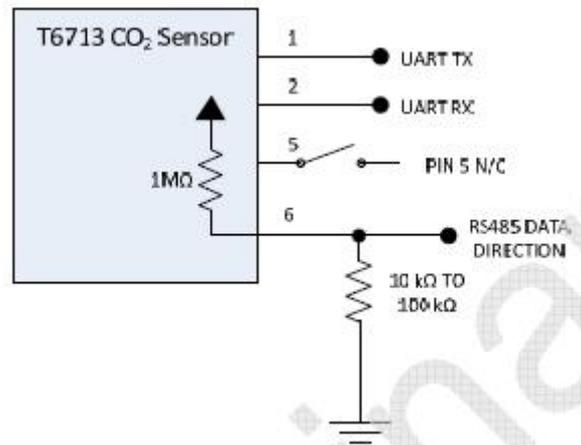


图 4 – 第三种 I/O 配置简图

表 3 – 第三种 I/O 针脚配置

针脚	说明
1	UART TX (传感器输出)
2	UART Rx (传感器输入)
5	变为供 Telaire 使用的“测试输入针”，用户须让其保持不连接状态。
6	变为输出针，用于驱动 RS485 收发器

上述第（3）种配置仅当 RS485 dropped 节点网络配置中使用了传感器的情况下有用。此类网络须由用户自供。在该配置下，6#针脚变为 RS-485 收发器的数据导向逻辑。

3 通信 – MODBUS 协议

T6713 系列传感器的所有通信均采用 Modbus 协议。文档《Modbus 串行线路协议及应用指南（版本号 1.02）》对 Modbus 协议进行了详细介绍。您可通过下列 Modbus 网址下载该文档：<http://www.modbus.org/specs.php>.

3.1 UART (RS232/RS485)

关于 UART 通信，请参考文档《Modbus 串行线路协议及应用指南（版本号 1.02）》中的建议。该文档包含了有关如何计算必需的 CRC（循环冗余检查）字节的详细信息。

需要注意的是，对基于串行线路（即 RS-232 和 RS-485）的 Modbus 通信，用户必须在 Modbus 请求的末尾包含循环冗余检查（CRC）字段。对基于 I²C 接口的通信，不需要进行 CRC 计算

3.2 I²C

I²C 类应用不需要使用串行线缆协议。传感器的确使用 Modbus 协议，不过会将消息封装成 I²C 格式。关于 I²C 通信的详细内容，请参考文档“I²C 技术规格及用户手册”，文档网址：www.nxp.com/documents/user_manual/UM10204.pdf。

**注：UART 和 I2C 协议均使用相同的默认地址"15'H ('21'D)

传感器始终作为从设备运行。在下面的示例中，采用 I²C 接口的执行器需要担当主发射器（Modbus 请求）或主接收器（Modbus 应答）与传感器通信。传感器不会发起任何通信，也就是说，传感器不会变成主设备和应答任何请求。应由主设备从传感器读取应答。

4 指令汇总

可以向传感器发送以下指令。

表 4 – Modbus 指令汇总

名称	Modbus 寄存器	寄存器地址	数据类型	说明
FIRMWARE_REVISION	输入 (RO)	'5001'D '1389'H	uint16_t	从传感器返回固件版本
STATUS	输入 (RO)	'5002'D '138A'H	uint16_t	从传感器返回一个状态寄存器。更多细节如下：
GAS PPM	输入 (RO)	'5003'D '138B'H	uint16_t	当前气体 ppm 浓度计算
RESET DEEVICE	线圈 (WO)	'1000'D '03E8'H	uint16_t	通过 Modbus 网络重置传感器
START SINGLE POINT CAL	线圈 (WO)	'1004'D '03EC'H	uint16_t	开始单点校准
SLAVE ADDRESS	保持 (RW)	'4005'D '0FA5'H	uin16_t	更改传感器地址（默认地址为'21'D ('15'H)

其中 RO=只读；WO=只写；RW=读写

以下章节提供了一些示例。

4.1 FIRMWARE REVISION 指令

该指令用于返回传感器当前的固件版本号。用户应通过 UART 给每一个指令附加 CRC。

利用 Modbus 的 Read Input Registers（读取输入寄存器）功能（4），并从地址'5001'D ("1389'H)读取一个寄存器。

示例：

UART (全部字节为 16 进制)

Modbus 请求 (UART)

'15'H	从地址
'04'H	功能代码
'13'H	起始地址(MSB)
'89'H	起始地址(LSB)
'00'H	寄存器数目 (MSB)
'01'H	寄存器数目 (LSB)
XX	CRC (LSB)
XX	CRC (MSB)

Modbus 应答 (UART)

'15'H	从地址
'04'H	功能代码
'02'H	字节数
XX	状态 (MSB)
XX	状态 (LSB)
XX	CRC (LSB)
XX	CRC (MSB)

示例 1 – 读取固件版本的 Modbus 请求/应答 (UART)

I²C (全部字节为十六进制)

默认的 I²C 从地址为‘21’D (‘15’H)，且没有显示。

Modbus 请求(主发射器/从接收器)

'04'H	功能代码
'13'H	起始地址(MSB)
'89'H	起始地址(LSB)
'00'H	待读取寄存器数目 (MSB)
'01'H	待读取寄存器数目 (LSB)

Modbus 应答(主接收器/从发射器)

'04'H	功能代码
'02'H	字节数
XX	状态 (MSB)
XX	状态 (LSB)

示例 2 – 读取固件版本的 Modbus 请求/应答 (I²C)

需要注意的是，传感器只能担当从设备。用户必须应用一个主 I²C 设备对传感器进行读写。

4.2 STATUS 指令

该指令返回一个包含传感器各种功能状态的寄存器。用户必须确认传感器无任何故障现象。同时，用户必须确认传感器已完成唤醒阶段。

利用 Modbus 的 Read Input Registers（读取输入寄存器）功能（4），并从地址 '5002'D（'138A'H）读取一个寄存器。

Modbus 请求(UART)

'15'H	从地址(默认为 21)
'04'H	功能代码
'13'H	起始地址 (MSB)
'8A'H	起始地址(LSB)
'00'H	待读取输入寄存器 (MSB)
'01'H	待读取输入寄存器 (LSB)
XX	CRC (LSB)
XX	CRC (MSB)

Modbus 应答 (UART)

'15'H	从地址
'04'H	功能代码
'02'H	字节数
XX	状态 (MSB)
XX	状态 (LSB)
XX	CRC (LSB)
XX	CRC (MSB)

示例 3 - 读取 STATUS（状态）寄存器的 Modbus 请求/应答 (UART)

Modbus 请求 (I²C)(主发射器/从接收器)

'04'H	功能代码
'13'H	起始地址 (MSB)
'8A'H	起始地址(LSB)
'00'H	待读取输入寄存器 (MSB)

Modbus 应答 (I²C) (主接收器/从发射器)

'04'H	功能代码
'02'H	字节数
XX	状态 (MSB)
XX	状态 (LSB)

 示例 4 - 读取 STATUS (状态) 寄存器的 Modbus 请求/应答(I²C)

STATUS (状态) 寄存器是一个位向量，其每一位代表的传感器的某些功能状态不是所有位都已分配。

表 5 - 状态寄存器逐位说明

位点	十六进制	说明
XXXXXXXX,XXXXXXXX1	'0001'H	故障状态
XXXXXXXX,XXXXXXXX1X	'0002'H	闪存故障
XXXXXXXX,XXXXX1XX	'0004'H	校准故障
XXXXXXXX,XXXX1XXX	'0003'H	不适用
XXXXXXXX,XXX1XXXX	'0010'H	不适用
XXXXXXXX,XX1XXXXX	'0020'H	不适用
XXXXXXXX,X1XXXXXX	'0040'H	不适用
XXXXXXXX,1XXXXXXXX	'0080'H	不适用
XXXXXXXX1,XXXXXXXX	'0100'H	RS-232
XXXXXX1X,XXXXXXXX	'0200'H	RS-485
XXXXX1XX,XXXXXXXX	'0400'H	I ² C
XXXX1XXX,XXXXXXXX	'0800'H	唤醒模式
XXX1XXXX,XXXXXXXX	'1000'H	不适用
XX1XXXXX,XXXXXXXX	'2000'H	不适用
X1XXXXXX,XXXXXXXX	'4000'H	不适用
1XXXXXXXX,XXXXXXXX	'8000'H	单点校准

对于故障状态，一个“1”表示有一个故障，一个“0”表示无故障。闪存故障属于严重故障（不可恢复）。校准故障可以通过再次运行校准程序并获得成功的校准结果而消除。

对于校准状态，一个“1”表示有一个校准周期正在执行过程中。当有一个校准周期正在执行时，不可启动另一个校准周期。这种启动新校准周期的请求会产生出错消息，通过 Modbus 应答进行报告。

若给传感器设置唤醒位，则传感器进入如下模式：内部寄存器进行初始化，且气体数据（ppm 浓度）不一定正确。

4.3 GAS PPM 指令

该指令用于报告当前气体 ppm 浓度测量值。

利用 Modbus 的 Read Input Registers（读取输入寄存器）功能（4），并从地址‘5003’D（‘I38B’H）读取一个寄存器。

Modbus 请求 (UART)

'15'H	从地址(默认为 21)
'04'H	功能代码
'13'H	起始地址(MSB)
'8B'H	起始地址(LSB)
'00'H	待读取输入寄存器 (MSB)
'01'H	待读取输入寄存器 (LSB)
XX	CRC (LSB)
XX	CRC (MSB)

Modbus 应答 (UART)

'15'H	从地址(默认为 21)
'04'H	功能代码
'02'H	字节数
XX	16 位数据的 MSB（最高有效位）
XX	16 位数据的 MSB（最低有效位）
XX	CRC (LSB)
XX	CRC (MSB)

示例 5 – 读取气体 ppm 浓度寄存器的 Modbu 请求/应答 (UART)

Modbus 请求 (I²C) (主发射器/从接收器)

'04'H	功能代码
'13'H	起始地址(MSB)
'8B'H	起始地址(LSB)
'00'H	待读取输入寄存器 (MSB)
'01'H	待读取输入寄存器 (LSB)

Modbus 应答 (I²C) (主接收器/从发射器)

'04'H	功能代码
'02'H	字节数
XX	16 位数据的 MSB（最高有效位）
XX	16 位数据的 MSB（最低有效位）



深圳市阿尔森物联技术有限公司

Shenzhen arsend IOT Dev.Co.,Ltd

16 位数据的 MSB (最低有效位)

示例 6 - 读取气体 ppm 浓度寄存器的 Modbu 请求/应答 (UART) (I²C)



执行以下步骤计算气体 ppm 浓度:

$$\text{ppm} = \text{MSB} * 256 + \text{LSB}$$

比如: 若 Modbus (UART) 应答为:

'15'H	从地址(默认为 21)
'04'H	功能代码
'02'H	字节数
'01'H	字节数
'9F'H	16 位数据的 MSB (最高有效位)
XX	16 位数据的 MSB (最低有效位)
XX	CRC (LSB)
	CRC (MSB)

示例 7- 计算气体 ppm 浓度

气体浓度计算如下:

$$1 * 256 + 159 = 415$$

其中:

$$'01'H = '1'D, '9F'H = '159'D$$

4.4 重置设备

传感器可以通过 Modbus 接口重置（通常不建议使用）。重置操作是即时生效的，且不会有任何应答。将电源关断重启后，传感器将开始工作。

利用 Modbus 的 Write Single Coil（写入单线圈寄存器）功能（5），并向位于地址‘1000’D (03E8’H)的寄存器写入‘00FF’H。

Modbus 请求 (UART)

'15'H	从地址(默认为 21)
'05'H	功能代码
'03'H	输出地址(MSB)
'E8'H	输出地址(LSB)
'FF'H	输出值(MSB)
'00'H	输出值(LSB)
XX	CRC (LSB)
XX	CRC (MSB)

示例 8 – 重置传感器的 Modbus 请求 (UART)

Modbus 请求 (I²C) (主发射器/从接收器)

'05'H	功能代码
'03'H	输出地址(MSB)
'E8'H	输出地址(LSB)
'FF'H	输出值(MSB)
'00'H	输出值(LSB)

示例 9 –重置传感器的 Modbus 请求(I²C)

注：不会有应答。重置操作即时生效。

4.5 START SINGLE POINT CALIBRATION 指令

该指令可启动单点校准程序。

单点校准程序通常在环境条件（~500ppm, 25 °C）下执行，从启动至完成校准的时间为数分钟（~6 分钟）。在此过程中可以查询其状态和当前气体 ppm 浓度读数。用户可以通过读取状态寄存器并注意是否设置有单点校准位来检查校准状态。允许中断校准过程。参见示例。

利用 Modbus 的 Write Single Coil（写入单线圈寄存器）功能 (5)，并向位于地址‘1004’D (03ECH)的寄存器写入‘00FF’H。校准过程中，写入‘0000’H 可中断校准功能。

Modbus 请求 (UART)

'15'H	从地址(默认为 21)
'05'H	功能代码
'03'H	输出地址(MSB)
'EC'H	输出地址(LSB)
'FF'H	输出值(MSB)
'00'H	输出值(LSB)
XX	CRC (LSB)
XX	CRC (MSB)

Modbus 应答 (UART)

'15'H	从地址(默认为 21)
'05'H	功能代码
'03'H	输出地址(MSB)
'EC'H	输出地址(LSB)
'FF'H	输出值(MSB)
'00'H	输出值(LSB)
XX	CRC (LSB)
XX	CRC (MSB)

示例 10 – 启动单点校准的 Modbus 请求/应答(UART)

Modbus 请求 (I²C)

'05'H	功能代码
'03'H	输出地址(MSB)
'EC'H	输出地址(LSB)
'FF'H	输出值(MSB)
'00'H	输出值(LSB)



深圳市阿尔森物联技术有限公司

Shenzhen arsend IOT Dev.Co.,Ltd

Modbus 请求 (I²C)

'05'H	功能代码
'03'H	输出地址(MSB)
'EC'H	输出地址(LSB)
'FF'H	输出值(MSB)
'00'H	输出值(LSB)

示例 11 -启动单点校准的 Modbus 请求/应答 (I²C)

校准过程需要数分钟。可通过读取状态寄存器确定单点状态。

校准指令不能二次启动，但可中断。详情示例如下。

Modbus 请求 (UART)

'15'H	从地址(默认为 21)
'05'H	功能代码
'03'H	输出地址(MSB)
'EC'H	输出地址(LSB)
'00'H	输出值(MSB)
'00'H	输出值(LSB)
XX	输出值(LSB)
XX	CRC (LSB)
	CRC (MSB)

Modbus 应答 (UART)

'15'H	从地址(默认为 21)
'05'H	功能代码
'03'H	输出地址(MSB)
'EC'H	输出地址(LSB)
'00'H	输出值(MSB)
'00'H	输出值(LSB)
XX	输出值(LSB)
XX	CRC (LSB)
	CRC (MSB)

示例 12 – 中断单点校准的 Modbus 请求/应答 (UART)

Modbus 请求 (I²C)

'05'H	功能代码
'03'H	输出地址(MSB)
'EC'H	输出地址(LSB)
'00'H	输出值(MSB)
'00'H	输出值(LSB)

Modbus 请求 (I²C)

'05'H	功能代码
'03'H	输出地址(MSB)
'EC'H	输出地址(LSB)
'00'H	输出值(MSB)
'00'H	输出值(LSB)

示例 13 - 中断单点校准的 Modbus 请求/应答(I²C)

注：没有显示基于 UART 的 Modbus 请求/应答。

4.6 CHANGE SLAVE ADDRESS 指令

允许更改 Modbus 的从地址。此更改仅在重置传感器或将电源关断重启后生效（比如通过 Modbus 接口）。此更改不会即时生效。

利用 Modbus 的 Write Single Register（写入单寄存器）功能（5），并向位于地址 '4005'D ('0FA5'H)的寄存器写入新地址。

本示例将当前从地址更改为'16'D ('10'H)。Modbus 请求 (UART)

'15'H	从地址(默认为 21)
'06'H	功能代码
'0F'H	寄存器地址(MSB)
'A5'H	寄存器地址 (LSB)
'00'H	寄存器值 (MSB) MSB 始终为 0
XX	寄存器值 (LSB) LSB 应处于 1-247 范围。
XX	CRC (LSB)
XX	CRC (MSB)

Modbus 应答 (UART)

'15'H	从地址(默认为 21)
'06'H	功能代码
'0F'H	寄存器地址(MSB)
'A5'H	寄存器地址 (LSB)
XX	寄存器值(MSB)
XX	寄存器值(LSB)
XX	CRC (LSB)
XX	CRC (MSB)

示例 14 – 更改默认从地址 (UART)

Modbus 请求 (I²C)

'06'H	功能代码
'0F'H	寄存器地址(MSB)
'A5'H	寄存器地址 (LSB)
'00'H	寄存器值 (MSB) MSB 始终为 0
XX	寄存器值 (LSB) LSB 应处于 1-247 范围。

Modbus 请求 (I²C)

'06'H	功能代码
'0F'H	寄存器地址(MSB)
'A5'H	寄存器地址 (LSB)
'00'H	寄存器值(MSB)
XX	寄存器值(LSB)

示例 15 – 更改默认从地址 (I²C)

注：此更改需重启传感器后生效：通过电源的关断重启或向 RESET（重置）寄存器写入'FF'H。