

用户手册

PCS09xx 系列

可编程电容传感控制器



1. 版本控制

1) 文档更新记录

日期	更新人	版本	备注
2015-6-4	ljiwen	V0.1.0	Initial
2015-6-22	Liur	V0.1.1	Add operation description
2015-8-26	Jacky	V0.1.2	Add command description
2016-05-13	Huangchen	V0.1.3	Update software UI
2018-06-23	Liur	V0.1.4	Migrate to PCS0902
2018-08-25	ljw	V0.1.5	Update outline dimension

目录

1	引言	5
1.1	知识产权保护申明	5
1.2	免责声明	5
2	概述	6
2.1	主要特性	6
2.2	功能特点	6
2.3	产品选型与订购信息	6
3	接口说明	7
3.1	接线端口位置	7
3.2	电源接口 J1	7
3.3	信号接口 J2	7
3.4	量测接口 J3	7
3.5	量测接口 J4	7
4	典型应用	7
4.1	单端模式 1	7
4.2	单端模式 2	8
4.3	单端模式 3	8
4.4	差分模式	9
4.5	触发配置	9
4.6	电容值校准	9
4.7	空针校准	11
4.8	自动上报模式	11
4.9	响应延时	11
5	安装方式	11
6	CANopen 通讯	12
6.1	CANopen 概述	12
6.2	CAN 帧结构	12
6.3	配置 CAN 通讯	13
6.3.1	节点 ID	13
6.3.2	波特率	13
6.4	系统信息获取	13
6.4.1	设备节点名称	13
6.4.2	硬件版本	14
6.4.3	软件版本	14
6.4.4	系统控制	14
6.5	传感器参数	14
6.5.1	传感器状态字	14
6.5.2	电容值	15
6.5.3	工作模式	15
6.5.4	触发设置	15
7	工具软件操作简介	16
7.1	安装准备	16
7.2	软件安装	16

7.2.1	驱动安装.....	16
7.2.2	工具软件安装.....	16
7.3	软件使用说明.....	16
7.3.1	使用准备.....	16
7.3.2	主界面.....	16
7.3.3	电容传感器控制.....	17
7.3.4	PDO 映射.....	18
7.3.5	固件升级.....	19
8	电气特性.....	19
9	安装尺寸图.....	19
10	附录一 PMC007CxSxP 对象字典表.....	21
11	附录二 CANOPEN 通讯示例.....	25
11.1	SDO 读写示例.....	25
11.1.1	SDO 读取.....	25
11.1.2	SDO 写入.....	26
12	附录三 PDO 配置示例.....	29
12.1	PDO 概述.....	29
12.1.1	PDO 的结构—映射参数.....	29
12.1.2	PDO 的结构—通信参数.....	30
12.1.3	PDO 触发方式.....	31
12.2	PDO 配置示例.....	32
13	附录四 SDO abort code error.....	32

1 引言

1.1 知识产权保护申明

PCS09xx 系列传感器已经申请如下国家专利：

- 控制器方案和方法已申请发明专利保护。
- 控制器电路已申请实用新型专利保护。
- 控制器外观已申请外观专利保护。

PCS09xx 系列传感器内嵌固件代码，任何试图破坏固件代码保护功能的行为均可视为违反了知识产权保护法案和条例。如果这种行为导致在未经 CQPUSI 授权的情况下，获取软件或其他受知识产权保护的成果，CQPUSI 有权依据该法案提起诉讼制止这种行为。

1.2 免责声明

本使用手册中所述的器件使用信息及其他内容仅为您提供便利，它们可能在未来版本中被更新。确保应用符合技术规范，是您自身应负的责任。CQPUSI 对这些信息不作任何形式的声明或担保，包括但不限于使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。CQPUSI 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 CQPUSI 器件用于生命维持和/或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 CQPUSI 免于承担法律责任和赔偿。

2 概述

2.1 主要特性

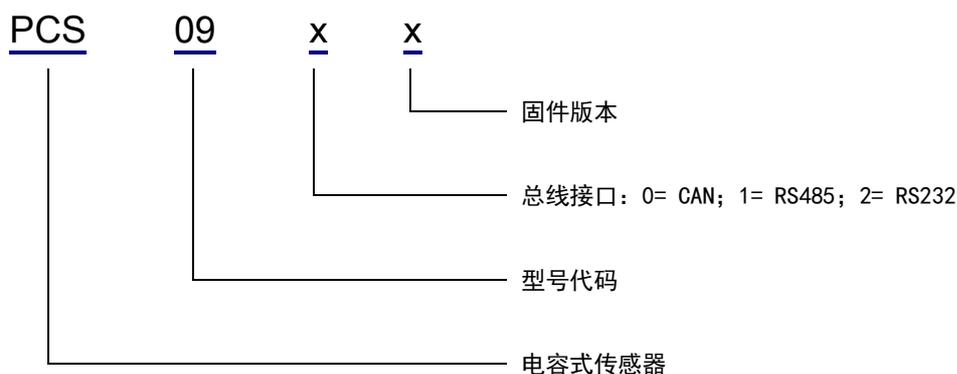
PCS09xx 是一种可编程电容传感控制器，其性价比高，接口简单灵活。该系列控制器能检测低于 1pF 的电容变化，并能将检测结果转换为 I/O 电平输出和直观的 LED 指示。PCS09xx 提供简单快捷的控制命令集，可通过 CAN（或 RS485 或 RS232）与控制主机相连接，直接读取测量值，适用于各类医疗、环保、实验室仪器的试剂液位检测、容器液位检测等应用场景。

2.2 功能特点

- ✓ 两个电容式传感器检测通道
- ✓ 边沿可配置数字电平输出
- ✓ LED 结果指示
- ✓ 电容检测范围 0 ~ 2.5nF
- ✓ 电容检测精度：10fF
- ✓ 微秒级超快响应速度
- ✓ 采用主动屏蔽技术提升检测可靠性
- ✓ 自适应低频激励电路实现超低 EMI 辐射
- ✓ 镀金屏蔽连接器提供优良的抗干扰能力
- ✓ 16mm x 56mm x 7.6mm 小体积利于安装
- ✓ 用户自定义可编程，并可配置离线自动运行
- ✓ 基于 VC++ 的控制例程和底层驱动

2.3 产品选型与订购信息

订购 PCS09xx 时请按以下格式提供具体的型号，以便我们准确及时的为您提供产品：



3 接口说明

3.1 接线端口位置



图 3-1

3.2 电源接口 J1

引脚	1	2
定义	VCC	GND

信号说明如下：

VCC：直流电源正极，5~12V；

GND：直流电源地；

3.3 信号接口 J2

引脚	1	2	3
定义	CANH	CANL	TR

信号说明如下：

CANH/CANL：CAN 通讯接口；

TR：触发信号输出引脚（3.3V），可配置为高电平或低电平触发；

3.4 量测接口 J3

电容量测通道二接口，采用 HRS U. FL-R-SMT-1 同轴连接器。该连接器接口的屏蔽层悬空，并尽可能靠近电极或试剂针；该连接器接口的芯线连接量测电极的正极。

3.5 量测接口 J4

电容量测通道一接口，采用 HRS U. FL-R-SMT-1 同轴连接器。该连接器接口的屏蔽层悬空，并尽可能靠近电极或试剂针；该连接器接口的芯线连接量测电极的正极。

4 典型应用

PCS09xx 控制器使用同轴电缆与待测电容极板连接，以下是双层金属探针液位检测的几种典型连接方式，可以通过 PUSICAN 工具软件中选择设置。

4.1 单端模式 1

此模式选择检测通道一工作，J4 的屏蔽层连接探针的外层金属，J4 的芯线连接探针的内层金属。由于 J4 的屏蔽层上有主动屏蔽信号，因此探针的外层金属必须与机架参考地或电源地保证绝缘隔离。

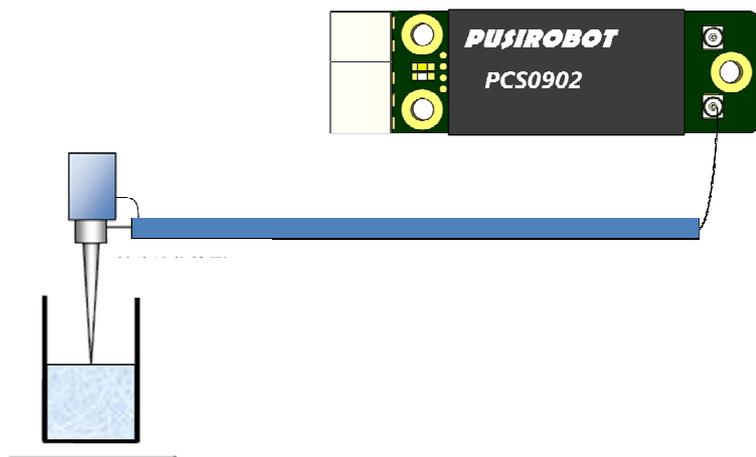


图 4-1

4.2 单端模式 2

此模式选择检测通道二工作，与单端模式 1 的连接方式类似。

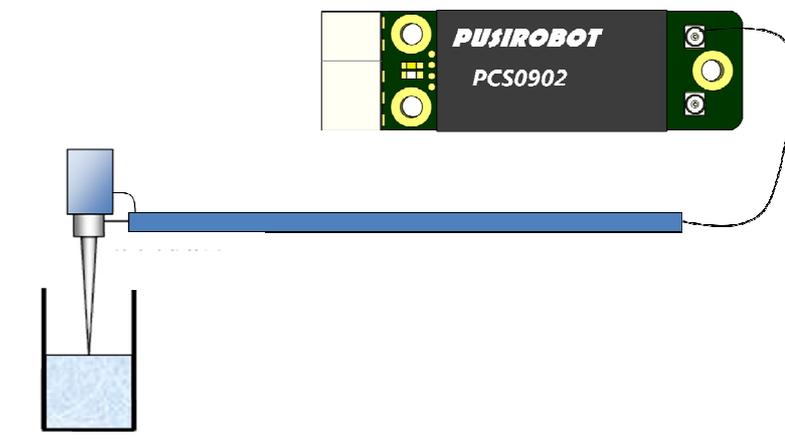


图 4-2

4.3 单端模式 3

当试剂瓶或待测液体的体积非常小，为了提高检测精度，可采用下图 3-6 的连接方法，其中 J4 电缆的芯线连接到探针，屏蔽层悬空不接；J3 电缆的芯线连接金属板（或金属容器）。该模式也可以用来量测电子元器件的两端电容值。

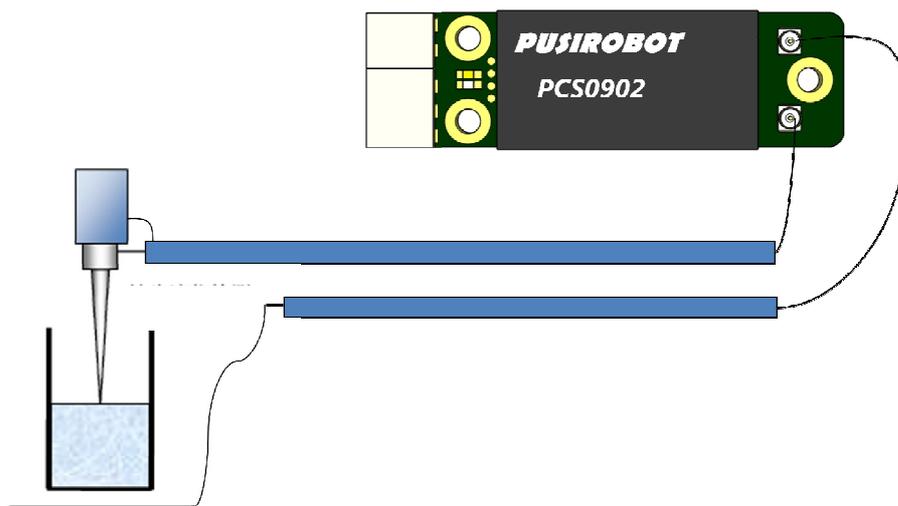


图 4-3

4.4 差分模式

如果设备机架的耦合电容非常大，或者需要对容器内的液位进行非接触式量测，可以采用差分模式。其中 J3 电缆的芯线连接到一侧电极，J4 电缆的芯线连接到另一侧的电极。两侧的电极如果采用双层金属，同时将外层金属分别连接到同轴电缆的屏蔽层，将会显著提高检测精度。

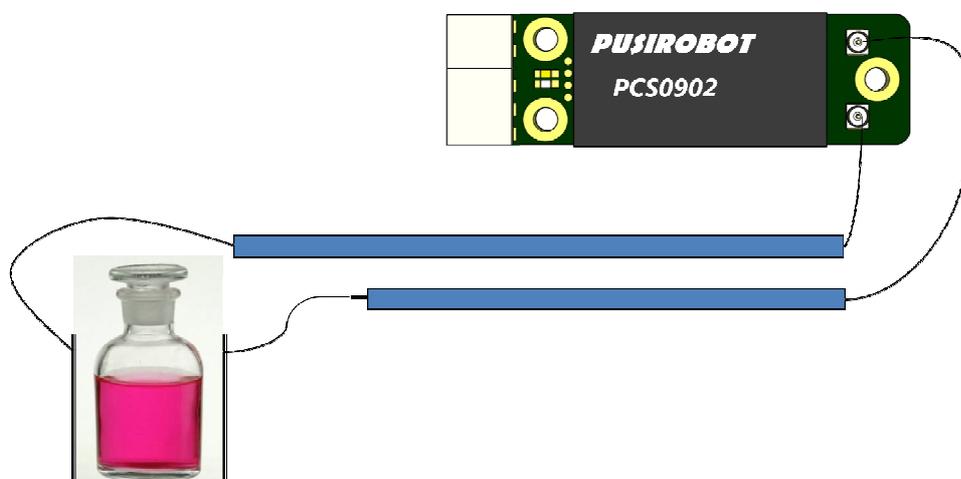


图 4-4

4.5 触发配置

PCS09xx 控制器支持电容值触发电平输出，触发的阈值均可通过调试工具软件 PUSICAN 配置。当检测的电容值达到阈值时，对应的 LED 指示灯亮，同时 TR 输出口就会输出对应电平。

4.6 电容值校准

为了获得更好的稳定性和检测精度，PCS09xx 控制器采用数字信号处理技术对量测环路进行控制，如下图 4-5 所示，系统外部环境（温度、湿度）和连接方式对系统参考电容值有显著影

响。

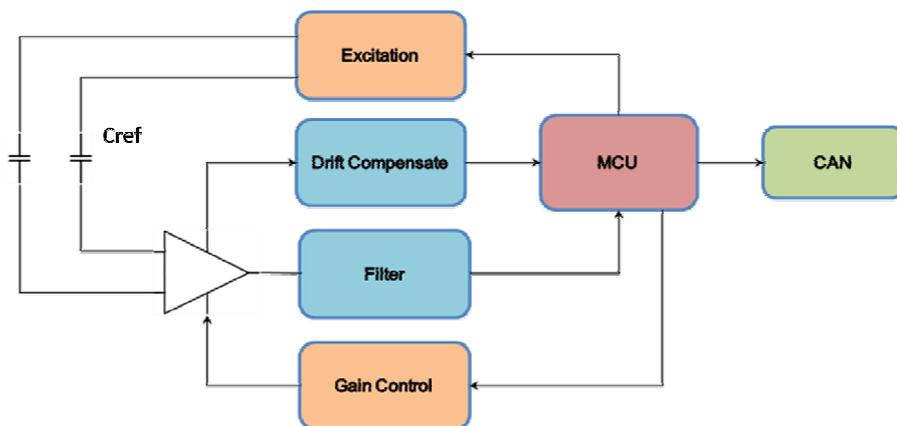


图 4-5

PCS09xx 内置的控制算法可以确保量测线性度（如图 4-6）最大程度免受这些环境因素的影响，但是用户可能需要视情形对基础电容值进行补偿。

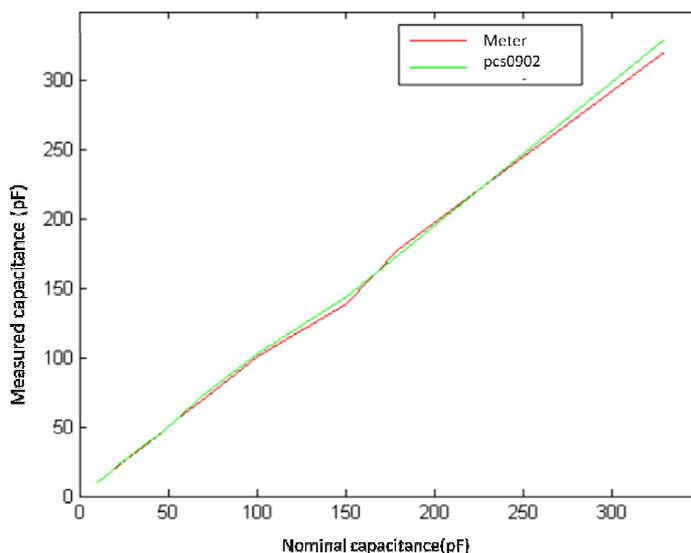


图 4-6

当在特定环境下对基础电容值进行补偿时，可以采用单端模式 3 连接一个标称陶瓷电容，如下图 4-7 所示，然后将标称值减去量测值，即可获得补偿值。注意典型的陶瓷电容误差值在 10%~30%之间。

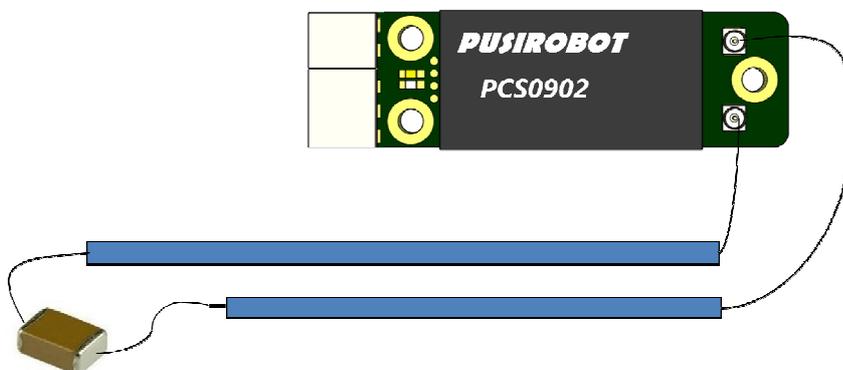


图 4-6

4.7 空针校准

当 PCS09xx 控制器和液位探针组件安装完成后, 机架等外部机构形成的基础电容已经稳定, 此时可以通过指令 (或 PUSICAN 软件工具) 执行空针校准, PCS09xx 将获取当前电容值, 并在后续的量测过程中自动减去该电容值。

4.8 自动上报模式

PCS09xx 控制器开机上电后, 自动进入连续测量模式, 板上的 Active 指示灯闪烁。量测结果将会通过 CANOPEN 协议的 PDO 方式自动上报。用户可以通过指令关闭自动上报功能。

注意: RS485/RS232 版本由于协议限制, 不具备自动上报功能。

4.9 响应延时

对于高精度实验室检测设备来说, 液位探测的精度、误报率、响应速度是三个重要的指标。PCS09xx 控制器采用高速处理器执行控制例程和滤波算法, 在确保检测精度和误报率的情况下, 将响应时间缩短到微秒级。如下图所示。同时, 为了减小试剂针带来的天线效应 (电感谐振量测法的主要缺陷之一), 降低电磁辐射, PCS09xx 控制器将激励频率限制在 10kHz 以内。

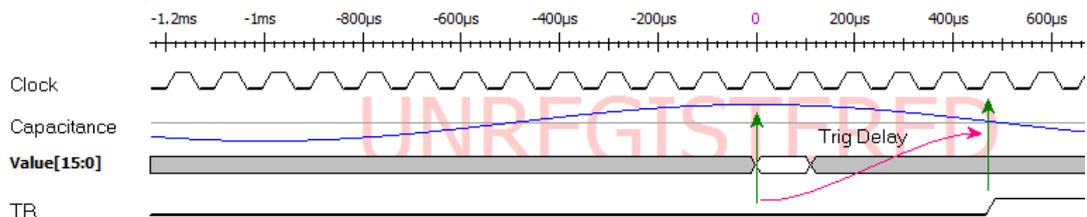


图 4-7

5 安装方式

PCS09xx 有三个安装孔, 建议采用三颗 M3 的铜柱辅助安装, 如下图 4-8 所示。铜柱的高度应大于 3mm, 以避免模块与基座干涉。一旦控制器被稳固锁定, 即可进行线缆连接。PCS09xx 采用弹簧端子连接器, 用户不需要对线缆进行端子压接, 直接将剥皮沾锡后的多芯线 (或者单芯铜线) 插入相应孔位即可, 适用的线规范围是 #26~#22AWG。

从连接器取出线头时, 可使用窄口螺丝刀或者我司提供的专用工具, 插入连接器上方的孔中, 挤开弹簧片即可。请勿强行拉扯线束退出, 以免造成弹簧片损坏。如下图 4-9。



图 4-8



图 4-9

6 CANopen 通讯

6.1 CANopen 概述

CAN 提供了所有的网络管理服务和报文传送协议，但并没有定义对象的内容或者正在通讯的对象的类型（它只定义了 how，没有定义 what），而这正是 CANopen 切入点。CANopen 是在 CAN 基础上开发的，使用了 CAN 通讯和服务协议子集，提供了分布式控制系统的一种实现方案。CANopen 在保证网络节点互用性的同时允许节点的功能随意扩展：或简单或复杂。

CANopen 的核心概念是设备对象字典（OD: Object Dictionary），在其它现场总线（Profibus, Interbus-S）系统中也使用这种设备描述形式。CANopen 通讯通过对象字典（OD）能够访问驱动器的所有参数。注意：对象字典不是 CAN 的一部分，而是在 CANopen 中实现的，PMC007CxSxP 支持的对象字典见附录一所示。

CANopen 通讯模型定义了如下几种报文（通讯对象）：

缩写	详称	说明
SDO	Service Data Object	用于非时间关键数据，比如参数。
PDO	Process Data Object	用于传输时间关键进程数据（给定值、控制字、状态信息等）。
SYNC	Synchronization Message	用于同步 CAN 节点。
EMCY	Emergency Message	用于传输驱动器的报警事件。
NMT	Network Management	用于 CANopen 网络管理。
Heartbeat	Error Control Protocol	用于监测所有节点的生命状态。

6.2 CAN 帧结构

CAN 通过数据帧在主机（控制器）和总线节点之间传输数据，下表为数据帧的结构。

帧头	仲裁域		控制域	数据域	校验域	应答域	帧尾
	COB-ID (通讯对象标识符)	RTR (远程请求)					
1位	11或29位	1位	6位	0~8字节	16位	2位	7位

软件实现收发时，上述帧结构中除了 COB-ID 和数据域，其他均由 CAN 收发控制器硬件完成，因此用户仅需要限定 COB-ID 和数据域。

注意：本驱动器采用标准帧格式，COB-ID 为 11 位，暂不支持远程帧。

COB-ID 的分配如下:

功能码				NODE ID (节点地址)						
10	9	8	7	6	5	4	3	2	1	0

控制器参数通过 SDO 读写对象访问, 对于需要实时上报到主站的驱动器状态信息可通过配置 PDO 实现。

6.3 配置 CAN 通讯

PMC007 出厂默认设置节点 ID 为 5, 波特率为 125Kbit/s, 用户使用时可以通过配套的 CANOPEN 主站调试工具对其进行修改设置。

6.3.1 节点 ID

对象名称	节点 ID
SDO ID	0x2002
对象类型	U8, rw
范围	1-127
存储类型	ROM
默认值	5

6.3.2 波特率

对象名称	波特率
SDO ID	0x2003
对象类型	U8, rw
范围	0, 1, 2, 3, 4, 5, 6, 7, 8
存储类型	ROM
默认值	5

各个取值对应的波特率如下:

- 0: 20Kbit/s
- 1: 25Kbit/s
- 2: 50Kbit/s
- 3: 100Kbit/s
- 4: 125Kbit/s
- 5: 250Kbit/s
- 6: 500Kbit/s
- 7: 800Kbit/s
- 8: 1000Kbit/s

6.4 系统信息获取

6.4.1 设备节点名称

对象名称	设备节点名称
SDO ID	0x1008

对象类型	string, ro
范围	-
存储类型	ROM
默认值	-

6.4.2 硬件版本

对象名称	硬件版本
SD0 ID	0x1009
对象类型	string, ro
范围	-
存储类型	ROM
默认值	-

6.4.3 软件版本

对象名称	软件版本
SD0 ID	0x100A
对象类型	string, ro
范围	-
存储类型	ROM
默认值	-

6.4.4 系统控制

对象名称	系统控制
SD0 ID	0x2007
对象类型	U8, ro
范围	1, 2, 3
存储类型	RAM
默认值	-

系统控制值的定义如下：

- 1: 跳转到 boot loader
- 2: 保存对象字典参数
- 3: 复位出厂设置

注意：对象字典中存储类型为 ROM 的参数通过 SD0 写入后被临时存到内存中，如需要永久保存，则需要执行掉电保存对象字典参数操作。

6.5 传感器参数

6.5.1 传感器状态字

对象名称	传感器状态字
SD0 ID	0x6000
对象类型	U16, rw
范围	bit
存储类型	RAM

默认值	0-3
-----	-----

传感器状态字每位定义如下：

Bit0: 触发

Bit1: 溢出

6.5.2 电容值

对象名称	电容值(单位: fF)
SD0 ID	0x6002
对象类型	S32, rw
范围	$-2^{31} \sim 2^{31}-1$
存储类型	RAM
默认值	0

6.5.3 工作模式

对象名称	工作模式
SD0 ID	0x6003
对象类型	U8, rw
范围	0~3
存储类型	RAM
默认值	0

工作模式的值定义如下：

0: 单端模式 1;

1: 单端模式 2;

3: 单端模式 3;

4: 差分模式;

6.5.4 触发设置

PCS009xx 可以设置触发输出，当实际电容值大于设置的电容值时，触发信号输出有效。

对象名称	触发设置
SD0 ID	0x6005
对象类型	Record
存储类型	ROM
参数个数	2

子索引 0x01: 触发极性设置

对象类型	U8, rw
范围	0~1
默认值	0

设置为 0 时，触发信号输出有效电平高电平，设置为 1 时，触发信号输出有效电平低电平。

子索引 0x02: 触发电容值

对象类型	S32, rw
范围	$-2^{31} \sim 2^{31}-1$

默认值	0
-----	---

7 工具软件操作简介

PCS09xx 可通过 CQPUSI 工具软件 PUSICAN 进行命令调试、IO 端口设置检测、步进电机参数调试和自定义编程等。

7.1 安装准备

工具软件 PUSICAN 需要 CAN 适配器 (USB2CAN 或 PCI2CAN) 的支持, 当前工具软件已经支持多种市面上常见的 USB2CAN 适配器, 如需要支持其它型号的适配器, 请联系销售人员。

7.2 软件安装

7.2.1 驱动安装

适配器驱动安装请按照适配器使用说明进行操作。

7.2.2 工具软件安装

调试工具 PUSICAN 属于绿色免安装软件, 下载后解压到专用文件夹内, 双击 `pusican.exe` 即可运行。

7.3 软件使用说明

7.3.1 使用准备

将 PCS09xx 和 CAN 适配器按错误! 未找到引用源。的方式连接到电脑, 然后给 PCS09xx 上电, 正常上电后 LED 灯将以 2.5Hz 的频率闪烁。

7.3.2 主界面

双击桌面 PUSICAN 快捷方式图标进入主界面。如图 7-1 所示:

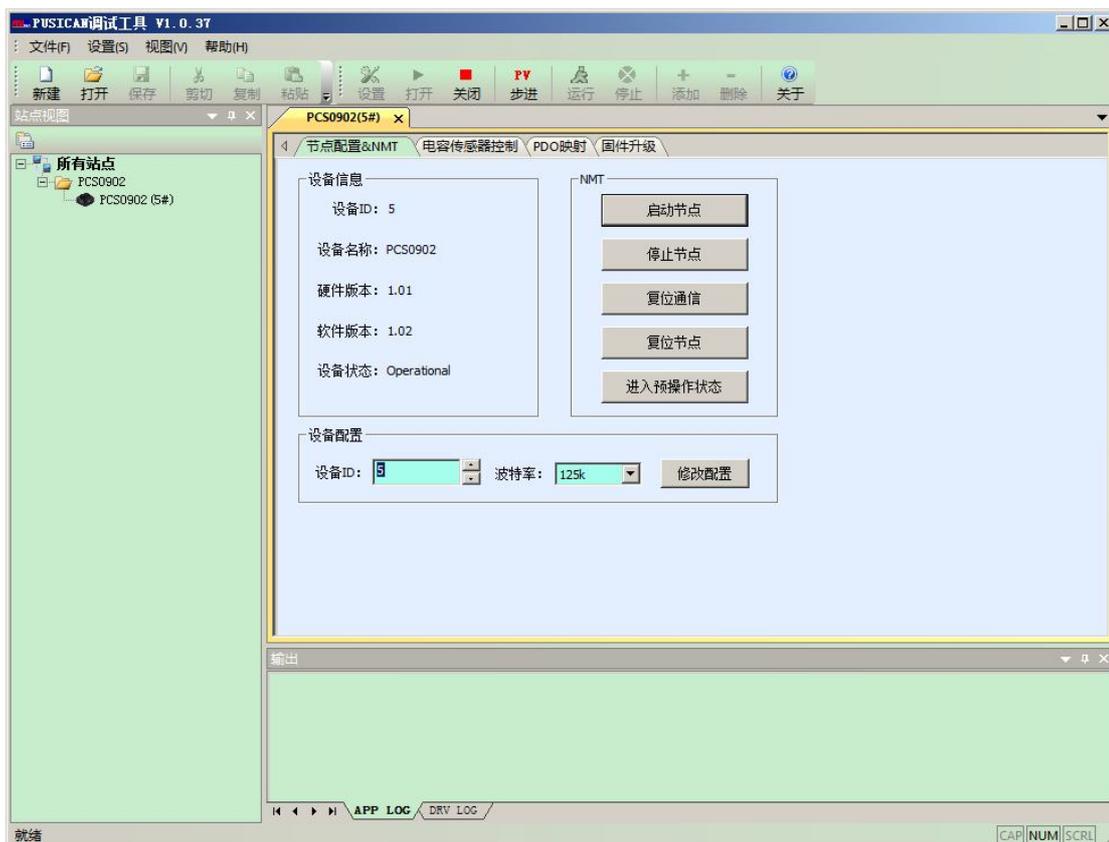


图 7-1

在主界面，点击“设置”图标选择适配器和波特率，PCS09xx 控制器出厂默认波特率为 125khz。点击“打开”图标，调试工具将启动适配器扫描站点，所有在线活动的设备均会列在左边的树形列表里，双击想要操作的站点，右边工作区将会显示该设备的控制界面。

启动节点：使 PCS09xx 进入可操作状态 (Operational)；

停止节点：使 PCS09xx 进入停止状态 (Stopped)，节点不再响应任何 SD0 命令；

复位通信：SD0 修改通信参数后可通过此操作使通信参数立即生效；

复位节点：通知节点重新进入复位上电流程；

预操作状态：在该状态，节点等待主站的网络命令，接收主站的配置请求，因此可以接收和发送除了 PDO 以外的所有报文；

7.3.3 电容传感器控制

在主界面点击“电容传感器控制”进入该界面，进入该界面后软件将自动获取电容值并才波形图方式显示在界面上，同时可对工作模式、触发模式等参数进行设置，如图 7-2 所示。

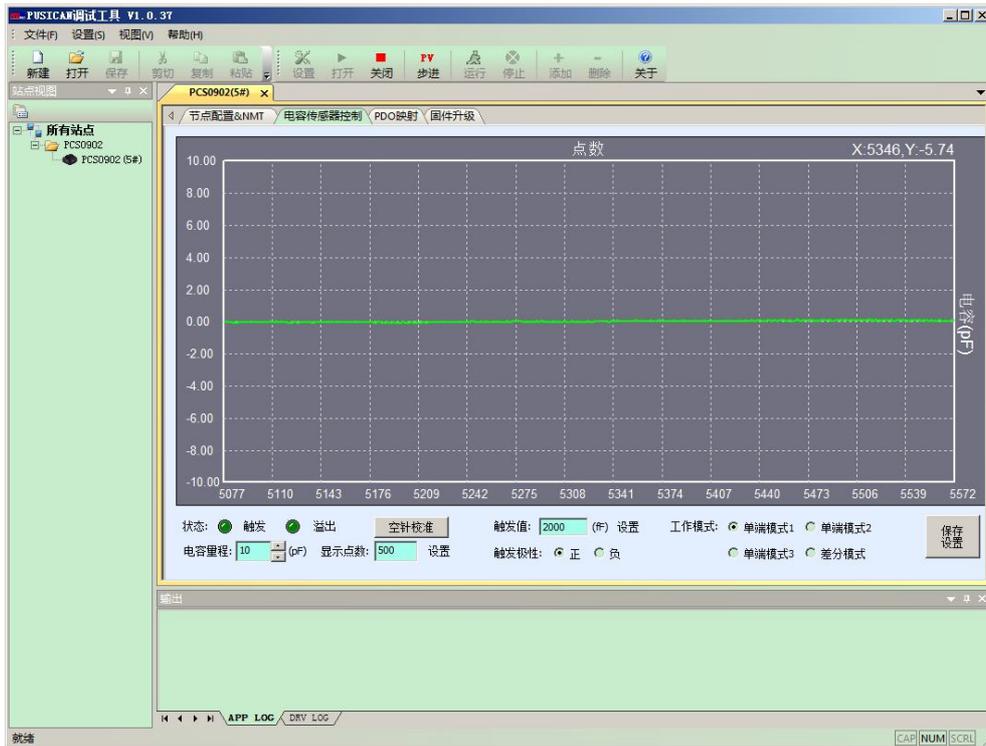


图 7-2

7.3.4 PDO 映射

在主界面点击“PDO 映射”进入该界面，进入该界面后软件将自动从设备读取当前的映射对象并显示在界面上，如 7-3 所示。

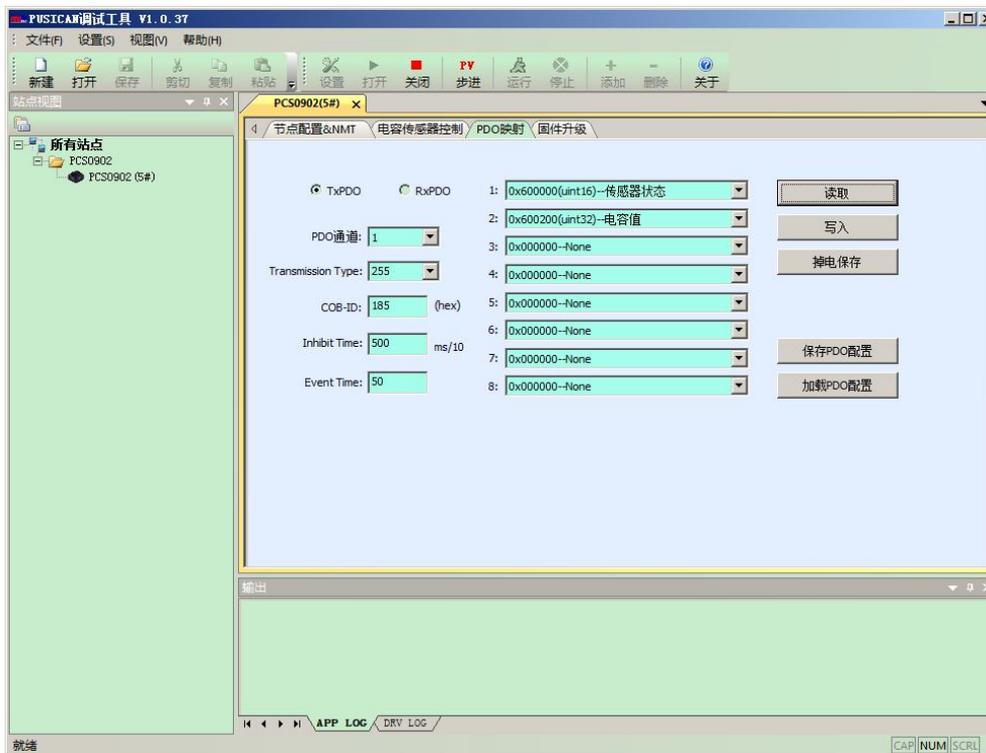


图 7-3

PCS09xx 支持 4 个通道的发送 PDO 和 4 个通道的接收 PDO，每个通道最多可映射 8 个对象。相关参数配置细节请参考附录二的 2.1 章节。

7.3.5 固件升级

PCS09xx 可通过 CAN 总线进行固件升级，固件升级在 boot loader 模式进行。在“固件升级”界面点击一次“进入 boot loader/应用”，PCS09xx 将进入 boot loader 模式，节点 ID 和波特率为应用模式下设置的节点 ID 和波特率。

进入 boot loader 模式后 LED 灯将双闪，在“应用路径”栏选择升级文件，点击“升级”按钮开始进行升级，如图 7-4 所示。升级完成后工具将弹出提示对话框，此时可以再点击一次“进入 boot loader/应用”（或者对控制器重新上电），控制器将切换到正常应用模式。

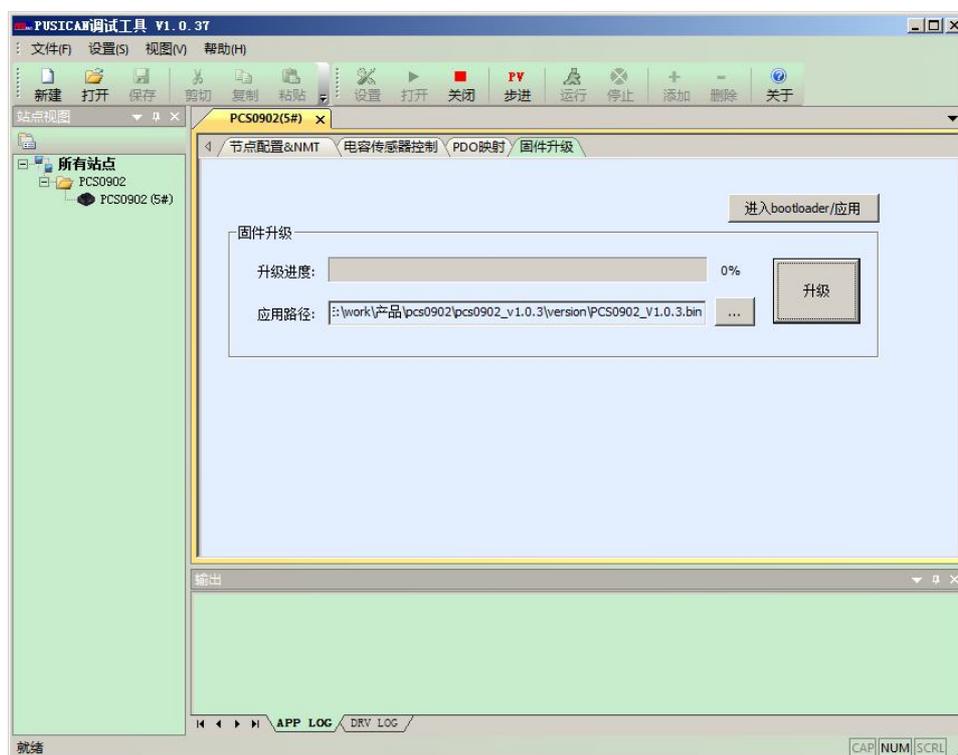
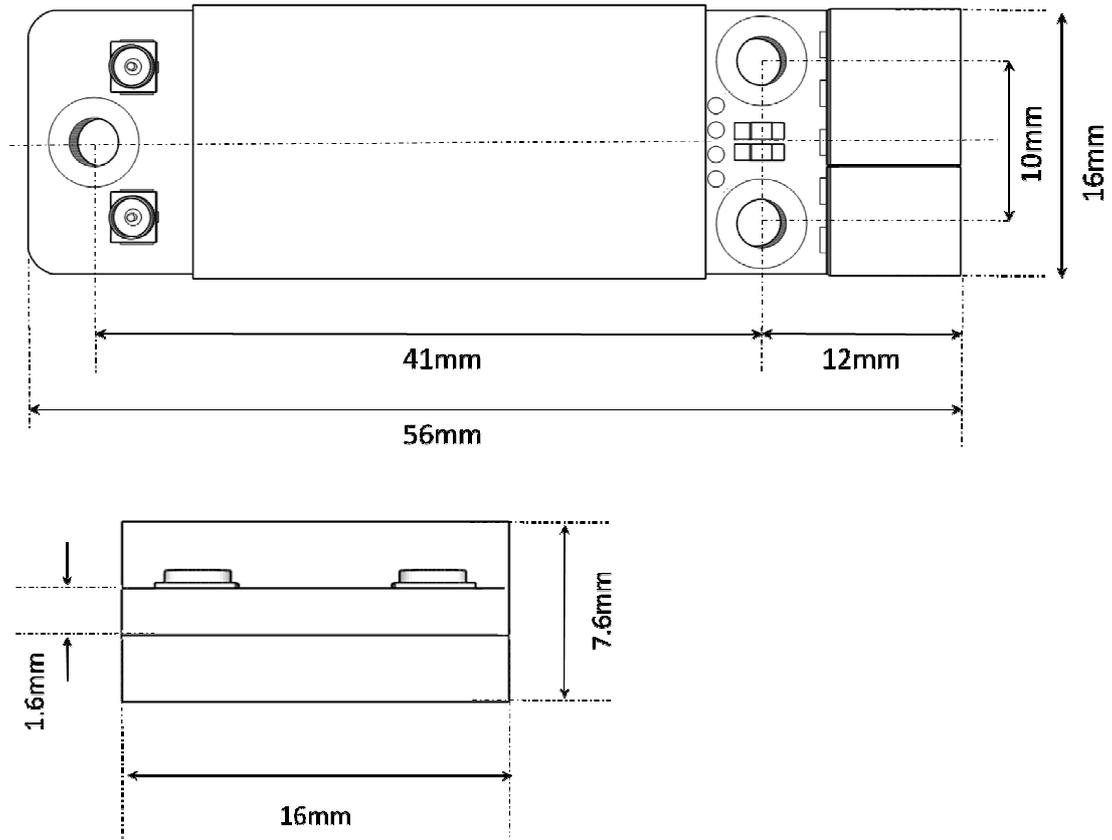


图 7-4

8 电气特性

参数	条件	最小值	典型值	最大值	单位
输入电压	常温 25°C	4.6	5	12	V
操作温度	12V 输入电压	-20		55	°C
最大工作电流	无触发情形	30	40	50	mA
量测接口高电平	常温 25°C	2.8	3.0	3.3	V
TR 接口低电平	12V 电源电压	-0.5	0	1.0	V
TR 接口高电平	12V 电源电压	2.8	3.0	3.3	V

9 安装尺寸图



10 附录一 PMC007CxSxP 对象字典表

Index 索引	Sub index 子索引	Object 对象类型	Name 名称	Type 数据类型	Attr. 属性	PDO 映射	存储 类型
1000h	--	VAR	Device type	UINT32	RO	NO	ROM
1001h	--	VAR	Error registor	UINT8	RO	Optional	RAM
1002h	--	VAR	manufacturer status register	UINT32	RO	Optional	RAM
1003h	--	ARRAY	pre-defined error field	--	--	--	RAM
	0h		number of errors	UINT8		NO	
	1h-7h		standard error field			Optional	
1005h	--	VAR	COB-ID SYNC	UINT32	RW	NO	ROM
1006h	--	VAR	communication cycle period	UINT32	RW	NO	ROM
1007h	--	VAR	synchronous window length	UINT32	RW	Optional	ROM
1008h	--	VAR	manufacturer device name	Visible String	const	NO	ROM
1009h	--	VAR	manufacturer hardware version	Visible String	const	NO	ROM
100ah	--	VAR	manufacturer software version	Visible String	const	NO	ROM
1014h	--	VAR	COB-ID Emergency message	UINT32	RO	NO	ROM
1015h	--	VAR	Inhibit Time EMCY	UINT16	RW	NO	ROM
1016h	--	ARRAY	Consumer Heartbeat Time	--	--	--	ROM
	0h		number entries	UINT8	RO	NO	
	1h-3h		Consumer Heartbeat Time	UINT32	RW	NO	
1017h	--	VAR	Producer Heartbeat Time	UINT16	RW	NO	ROM
1018h	--	RECORD	Identity Object	--	--	--	ROM
	0h		number of entries	UINT8	RO	NO	
	1h		Vendor ID	UINT32	RO	NO	
	2h		Product code	UINT32	RO	NO	
	3h		Revision number	UINT32	RO	NO	
	4h		Serial number	UINT32	RO	NO	
1200h	--	RECORD	Server SDO parameter	--	--	--	ROM
	0h		number of entries	UINT8	RO	NO	
	1h		COB-ID Client->Server (rx)	UINT32	RO	NO	

	2h		COB-ID Server -> Client (tx)	UINT32	RO	NO	
	3h		Node-ID of the SDO client	UINT32	RW	NO	
1280h	--	RECORD	Client SDO parameter	--	--	--	RAM
	0h		number of entries	UINT8	RO	NO	
	1h		COB-ID Client->Server (tx)	UINT32	RW	NO	
	2h		COB-ID Server -> Client (rx)	UINT32	RW	NO	
	3h		Node-ID of the SDO server	UINT32	RW	NO	
1400h	--	RECORD	receive PDO parameter	--	--	--	ROM
	0h		largest sub-index supported	UINT8	RO	NO	
	1h		COB-ID used by PDO	UINT32	RW	NO	
	2h		transmission type	UINT8	RW	NO	
	3h		inhibit time	UINT16	RW	NO	
	4h		compatibility entry	UINT8	RW	NO	
	5h		event timer	UINT16	RW	NO	
	1401h	--	RECORD	receive PDO parameter	--	--	--
0h			largest sub-index supported	UINT8	RO	NO	
1h			COB-ID used by PDO	UINT32	RW	NO	
2h			transmission type	UINT8	RW	NO	
3h			inhibit time	UINT16	RW	NO	
4h			compatibility entry	UINT8	RW	NO	
5h			event timer	UINT16	RW	NO	
1402h		--	RECORD	receive PDO parameter	--	--	--
	0h		largest sub-index supported	UINT8	RO	NO	
	1h		COB-ID used by PDO	UINT32	RW	NO	
	2h		transmission type	UINT8	RW	NO	
	3h		inhibit time	UINT16	RW	NO	
	4h		compatibility entry	UINT8	RW	NO	
	5h		event timer	UINT16	RW	NO	
	1403h	--	RECORD	receive PDO parameter	--	--	--
0h			largest sub-index supported	UINT8	RO	NO	
1h			COB-ID used by PDO	UINT32	RW	NO	
2h			transmission type	UINT8	RW	NO	
3h			inhibit time	UINT16	RW	NO	

	4h		compatibility entry	UINT8	RW	NO	
	5h		event timer	UINT16	RW	NO	
1600h	--	RECORD	receive PDO mapping	--	--	--	ROM
	0h		number of mapped application objects in PDO	UINT8	RO	NO	
	1h-8h		PDO mapping for the nth application object to be mapped	UINT32	RW	NO	
1601h	--	RECORD	receive PDO mapping	--	--	--	ROM
	0h		number of mapped application objects in PDO	UINT8	RO	NO	
	1h-8h		PDO mapping for the nth application object to be mapped	UINT32	RW	NO	
1602h	--	RECORD	receive PDO mapping	--	--	--	ROM
	0h		number of mapped application objects in PDO	UINT8	RO	NO	
	1h-8h		PDO mapping for the nth application object to be mapped	UINT32	RW	NO	
1603h	--	RECORD	receive PDO mapping	--	--	--	ROM
	0h		number of mapped application objects in PDO	UINT8	RO	NO	
	1h-8h		PDO mapping for the nth application object to be mapped	UINT32	RW	NO	
1800h	--	RECORD	transmit PDO parameter	--	--	--	ROM
	0h		largest sub-index supported	UINT8	RO	NO	
	1h		COB-ID used by PDO	UINT32	RW	NO	
	2h		transmission type	UINT8	RW	NO	
	3h		inhibit time	UINT16	RW	NO	
	4h		reserved	UINT8	RW	NO	
	5h		event timer	UINT16	RW	NO	
1801h	--	RECORD	transmit PDO parameter	--	--	--	ROM
	0h		largest sub-index supported	UINT8	RO	NO	

	1h		COB-ID used by PDO	UINT32	RW	NO	
	2h		transmission type	UINT8	RW	NO	
	3h		inhibit time	UINT16	RW	NO	
	4h		reserved	UINT8	RW	NO	
	5h		event timer	UINT16	RW	NO	
1802h	--	RECORD	transmit PDO parameter	--	--	--	ROM
	0h		largest sub-index supported	UINT8	RO	NO	
	1h		COB-ID used by PDO	UINT32	RW	NO	
	2h		transmission type	UINT8	RW	NO	
	3h		inhibit time	UINT16	RW	NO	
	4h		reserved	UINT8	RW	NO	
	5h		event timer	UINT16	RW	NO	
1803h	--	RECORD	transmit PDO parameter	--	--	--	ROM
	0h		largest sub-index supported	UINT8	RO	NO	
	1h		COB-ID used by PDO	UINT32	RW	NO	
	2h		transmission type	UINT8	RW	NO	
	3h		inhibit time	UINT16	RW	NO	
	4h		reserved	UINT8	RW	NO	
	5h		event timer	UINT16	RW	NO	
1a00h	--	RECORD	transmit PDO mapping	--	--	--	ROM
	0h		number of mapped application objects in PDO	UINT8	RO	NO	
	1h-8h		PDO mapping for the n-th application object to be mapped	UINT32	RW	NO	
1a01h	--	RECORD	transmit PDO mapping	--	--	--	ROM
	0h		number of mapped application objects in PDO	UINT8	RO	NO	
	1h-8h		PDO mapping for the n-th application object to be mapped	UINT32	RW	NO	
1a02h	--	RECORD	transmit PDO mapping	--	--	--	ROM
	0h		number of mapped application objects in PDO	UINT8	RO	NO	

	1h-8h		PDO mapping for the n-th application object to be mapped	UINT32	RW	NO	
1a03h	--	RECORD	transmit PDO mapping	--	--	--	ROM
	0h		number of mapped application objects in PDO	UINT8	RO	NO	
	1h-8h		PDO mapping for the n-th application object to be mapped	UINT32	RW	NO	
2002h	--	VAR	节点 ID	UINT8	RW	NO	ROM
2003h	--	VAR	波特率	UINT8	RW	NO	ROM
6000h	--	VAR	传感器状态字	UINT16	RW	Optional	RAM
6001h	--	VAR	控制器控制字	UINT16	RW	Optional	RAM
6002h	--	VAR	电容值	INT32	RW	NO	RAM
6003h	--	VAR	工作模式	UINT8	RW	NO	RAM
6005h	--	RECORD	触发设置	UINT8	RW	NO	ROM
	0h		参数个数	UINT8	RO	NO	ROM
	1h		触发输出极性	UINT8	RW	NO	RAM
	2h		触发值	INT32	RW	NO	RAM

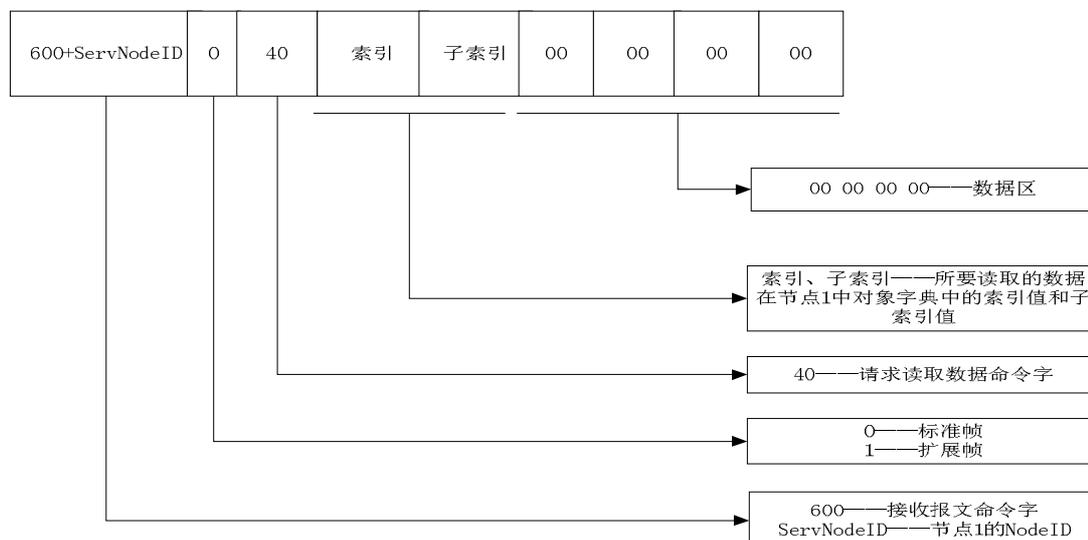
11 附录二 CANOPEN 通讯示例

11.1 SDO 读写示例

11.1.1 SDO 读取

11.1.1.1 数据帧格式

主站发送：



从站响应:

数据长度为 1 个字节时								
580+ServNodeID	0	4F	索引	子索引	d0	0	0	0
数据长度为 2 个字节时								
580+ServNodeID	0	4B	索引	子索引	d0	d1	0	0
数据长度为 3 个字节时								
580+ServNodeID	0	47	索引	子索引	d0	d1	d2	0
数据长度为 4 个字节时								
580+ServNodeID	0	43	索引	子索引	d0	d1	d2	d3

11.1.1.2 SDO 读取示例

主站发送: 605 40 02 60 00 00 00 00 00

从站响应: 585 43 02 60 00 FC 5E 00 00

主站向节点 ID 为 5 的设备发起读请求, 请求的索引地址为 0x6002, 子地址为 0x00, 对应 PCS09xx 对象字典中的电容值。从站响应 43 表示该参数长度为四个字节, 数据为 0x5EFC, 表示电容值为 24316fF。

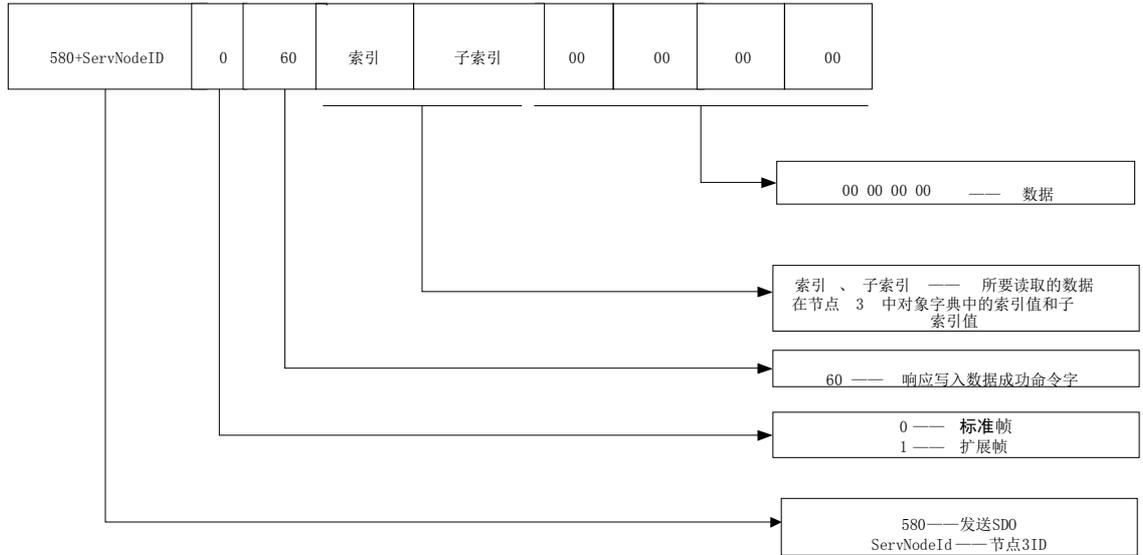
11.1.2 SDO 写入

11.1.2.1 数据帧格式

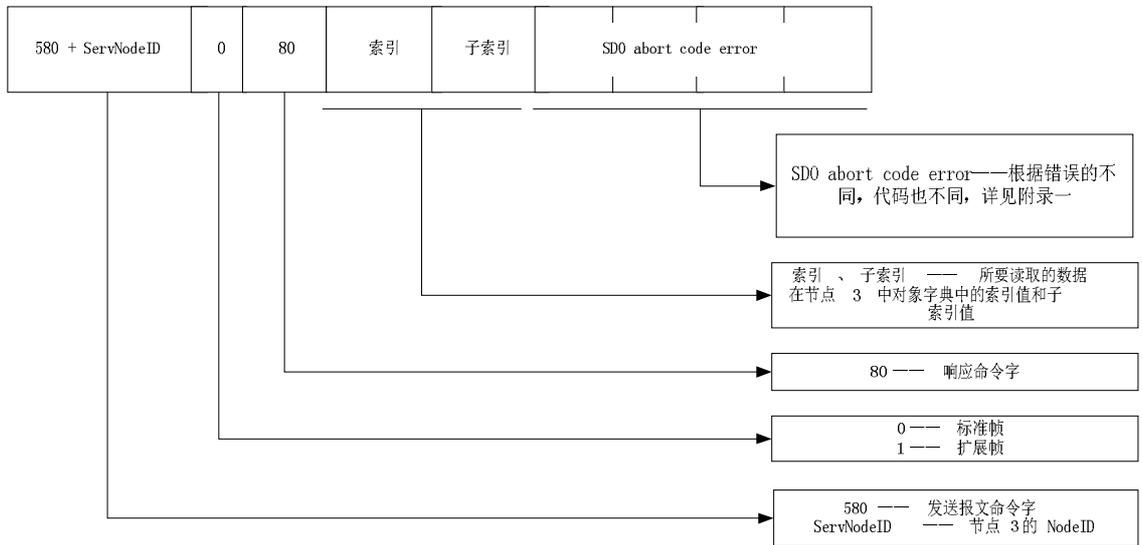
主站发送:

数据长为 1 个字节时								
600+ServNodeID	0	2F	索引	子索引	d0	0	0	0
数据长度为 2 个字节时								
600+ServNodeID	0	2B	索引	子索引	d0	d1	0	0
数据长度为 3 个字节时								
600+ServNodeID	0	27	索引	子索引	d0	d1	d2	0
数据长度为 4 个字节时								
600+ServNodeID	0	23	索引	子索引	d0	d1	d2	d3

从站正确响应:



从站错误响应:



注：其中 SDO abort code error 根据具体错误返回相应的参数，其具体参数详见附录二。

11.1.2.2 SDO 写入示例

主站发送：605 2F 03 20 00 07 00 00 00

从站响应：585 60 03 20 00 00 00 00 00

主站向节点 ID 为 5 的设备发起写请求，请求的索引地址为 0x2003, 子地址为 0x00, 写入的数据为 7, 对应 PMC007 对象字典中的波特率设置参数，写入数据 7 表示设置波特率为 800Kbit/s。从站响应 60 表示写入成功。

主站发送：605 23 04 60 00 80 00 00 00

从站响应：585 80 04 60 00 22 00 00 08

主站向节点 ID 为 5 的设备发起写请求，请求的索引地址为 0x6004, 子地址为 0x00, 写入的数据为 0xC80 (3200), 对应 PMC007 对象字典中的步进命令，写入数据 3200 表示让电机执行 3200 步步进。从站响应 80 表示写入失败，错误码为 0x08000022, 查看附录三可知

该错误码表示由于当前设备状态导致数据不能传送或保存到应用。查看控制器的状态参数是否外部停止有效和错误状态是否有错误。

12 附录三 PDO 配置示例

12.1 PDO 概述

PDO 通信是基于生产者/消费者 (Producer/Consumer) 模型, 主要用于传输实时数据。产生数据的节点将带有自己节点号的数据放到总线上, 需要该数据的节点可以配置为接收该节点发送的数据。PDO 的传输是由事件触发的, 这样的事件可以是代表一个 PDO 变量的变化; 可以是时间的过期或者是接收到一个特定的消息。过程数据直接在一个 CAN 消息中传输而不需要协议头文件。一个 PDO 的长度是在 0 到 8 个字节之间。

PDOs 包含在其映射参数和通信参数中, PMC007CxSxP 支持 4 个 PDOs。

12.1.1 PDO 的结构—映射参数

在对象词典中一个 PDO 由临近的条目构成。所谓的映射参数定义了这些条目的连接。一个映射参数通过索引、分索引和位数定义数据源。

例如:

<i>Index</i>	<i>Sub-index</i>	<i>Object Data</i>	<i>Description</i>
0x1A00	0	4	Number of mapped entries
	1	0x20000310	The entry at index 0x2000, sub-index 3, with a length of 16 bit, is mapped to bytes 0 and 1 within the CAN message.
	2	0x20000108	The entry at index 0x2000, sub-index 1, with a length of 8 bit, is mapped to byte 2 within the CAN message.
	

Table 1: Example for mapping parameters for the first TPDO

一个 CAN 消息最多有 8 个字节。这意味着当使用一个 PDO 时, 在一个 PDO 中最多发送 8 个字典的对象条目。

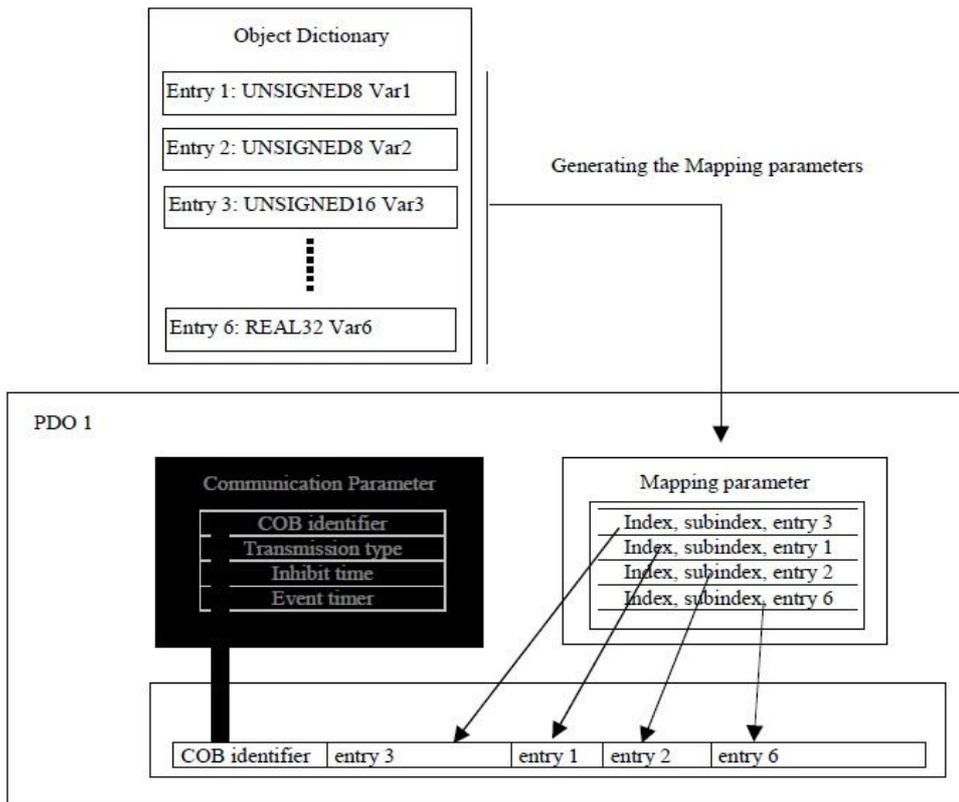


Figure 3: Mapping of Object Dictionary entries into a PDO

12.1.2 PDO 的结构—通信参数

为了传输一个 PDO，通信参数定义了传输的性质和 CAN 标识符(CAN identifier)。

Index	Sub-index	Object Data	Description
1800h	0	Number on entries	
	1	COB-ID	CAN identifier for the PDO
	2	Transmission Type	transmission type of the PDO
	3	Inhibit Time	minimum inhibit time for a TPDO
	4	reserved	reserved
	5	Event Time	maximum time between two TPDOs

Table 4: Communication parameter for the first TPDO

PDO 通信参数是在对象词典的条目

(RPDOs: index 0x1400–0x15FF, TPDOs: 0x1800–0x19FF)

如果允许，映射参数在数据服务对象帮助下可以通过 CAN 修改。

12.1.2.1 COB-ID(CAN 标识符, 分索引 1)

COB-ID 作为身份证明，PDO 的优先权在总线入口(access)之前。对于每一个 CAN 消息来说只允许有一个发送者(生产者)。然而，对这个已存在的消息来说其允许多个接收者(消费者)。

Bit	31	30	29	28 - 11	10 - 0
11-bit-ID	0/1	0/1	0	00000000000000000000	11-bit identifier
29-bit-ID	0/1	0/1	1	29-bit identifier	

Table 5: Structure of a COB-ID for PDOs

第 30 位为 0，表示对这个 PDO 来说一个远程发送请求（RTR）是被允许。

PDO COB-ID 分配：

PDO1(发送)	181H-1FFH
PDO1(接收)	201H-27FH
PDO2(发送)	281H-2FFH
PDO2(接收)	301H-37FH
PDO3(发送)	381H-3FFH
PDO3(接收)	401H-47FH
PDO4(发送)	481H-4FFH
PDO4(接收)	501H-57FH

12.1.3 PDO 触发方式

PDO 的发送可通过以下方式触发：

- 1) 事件触发；
- 2) 时间性触发；
- 3) 单个查询；
- 4) 同步；

当单独使用事件触发 PDO 发送时，一旦过程发生改变则发送 PDO。这可能带来非常严重的后果，那就是当某个过程数据变化的频率非常高时，这个 PDO 不停地发送，导致其它节点的报文发送不出去，严重影响总线的效率。

CANopen 采用 “Inhibit time” 机制来解决这个问题。Inhibit time 是一个可以配置的以 100us 为单位的时间段。相同 PDO 发送至少间隔这个时间段，这样就可以确定某个事件触发的 PDO 的最大发送频率。

一般来说，PDO 发送可以由任何触发方式的组合来触发。但最常用的是将事件触发和时间触发结合起来使用。单独事件触发，过程数据长时间没有变化（比如温度变量等），该 PDO 长时间没被触发，这样会影响刚加入网络中的节点，这时如果再加上时间触发的方式，就可以强制 PDO 在规定的时间内再发送一次。例如某个 PDO 配置 Inhibit time 为 50, Event timer 为 250，这样该 PDO 可以在过程数据变化时发送，发送的最小间隔是 5ms，另一方面不管数据有没有发生变化，每隔 250ms PDO 都会发送一次。

PDO 触发方式通过配置 PDO 通信参数对象字典的子索引 2 配置实现。该索引的取值范围为 0-255。下面列出了不同的值对就不同的触发方式。

- 0: PDO 在收到 SYNC 后发送, 但不是周期的;
- 1-240: PDO 在收到 SYNC 后发送, 周期发送, 该值为两次发送 PDO 之间的间隔的 SYNC 个数;
- 255: 事件触发;

12.2 PDO 配置示例

PMC007CxSxP 支持通过 SDO 配置 PDO 映射, 以配置 GPIO 值为 TPD01 为示例, 发送的 SDO 为:

设置通信的 COB-ID 为 187, 即节点 ID 为 7 的设备接收该 PDO

主站发送: 605 23 00 18 01 87 01 00 00

设置为事件触发

主站发送: 605 2F 00 18 02 FF 00 00 00

设置 Inhibit time 为 5ms

主站发送: 605 2B 00 18 03 32 00 00 00

设置 Event time 为 1000ms

主站发送: 605 2B 00 18 05 E8 03 00 00

设置映射条目数 1

主站发送: 605 2F 00 1A 00 01 00 00 00

设置映射参数, 将 0x6002 映射到 TPD01

主站发送: 605 23 00 1A 01 20 00 02 60

配置完成后 PCS09xx 会每隔 1s 发出 PDO 报文, 报文中包含电容值信息。

13 附录四 SDO abort code error

中止代码	代码功能描述
05030000	触发位没有交替改变
05040000	SDO 协议超时
05040001	非法或未知的 Client/Server 命令字
05040002	无效的块大小 (仅 Block Transfer 模式)
05040003	无效的序号 (仅 Block Transfer 模式)
05030004	CRC 错误 (仅 Block Transfer 模式)
05030005	内存溢出
06010000	对象不支持访问
06010001	试图读只写对象
06010002	试图写只读对象
06020000	对象字典中对象不存在
06040041	对象不能够映射到 PDO
06040042	映射的对象的数目和长度超出 PDO 长度
06040043	一般性参数不兼容
06040047	一般性设备内部不兼容
06060000	硬件错误导致对象访问失败
06060010	数据类型不匹配, 服务参数长度不匹配
06060012	数据类型不匹配, 服务参数长度太大

06060013	数据类型不匹配，服务参数长度太短
06090011	子索引不存在
06090030	超出参数的值范围（写访问时）
06090031	写入参数数值太大
06090032	写入参数值太小
06090036	最大值小于最小值
08000000	一般性错误
08000020	数据不能传送或保存到应用
08000021	由于本地控制导致数据不能传送或保存到应用
08000022	由于当前设备状态导致数据不能传送或保存到应用
08000023	对象字典动态产生错误或对象字典不存在