

USB接口适配器评估模块

用户指南

文献编号: SLLU093

2006年8月



前言.....	7
1 介绍	9
1.1 操作硬件项目清单.....	9
1.2 操作软件项目列表.....	10
1.3 操作和故障排除的分步说明.....	10
2 硬件设计总结	11
2.1 介绍.....	11
2.2 带状电缆连接器J2和引脚排列.....	11
2.3 引脚8、9和的上拉电阻配置 10	12
2.4 电可擦除只读存储器.....	12
2.5 USB连接的LED指示.....	12
3 固件编程和通信协议	13
3.1 固件功能摘要	13
3.2 固件版本命令和响应.....	13
3.3 I ² C写入命令和响应.....	14
3.4 I ² C读取命令和响应.....	14
3.5 读/写端口0命令和响应.....	15
3.6 EEPROM程序命令和响应.....	15
3.7 EEPROM读取命令和响应.....	16
3.8 设置上拉电阻器命令和响应.....	16
3.9 I ² C-PMBUS-SMBUS速度命令与响应设置.....	17
3.10 通用I ² C写入命令和响应.....	17
3.11 通用I ² C读取命令和响应.....	18
3.12 板测试命令和响应.....	18
3.13 其他用于不同应用程序的命令.....	18
4 固件版本方案	19
4.1 介绍.....	19
4.2 实现合规自定义固件的规则.....	20
4.3 符合要求的版本控制.....	20
5 GUI和PC库	25
5.1 视窗操作系统.....	26
5.2 设备传输.....	27
5.3 GUI应用程序.....	28
6 示意图、物料清单、PCB布局	29
6.1 示意图.....	30
6.2 物料清单 (BOM)	31
6.3 印刷电路板 (PCB) 布局.....	33
A 通信协议列表	37
A.1 主机PC到USB接口适配器命令.....	37
SLLU093-2006年6月 USB接口适配器向主机PC发送响应命令.....	38
B PMBUS/SMBUS通信	39

B.1	概述.....	39
-----	---------	----

B.2	支持的基本SMBUS事务类型.....	39
B.3	用于PMBUS/SMBUS通信的特殊信号.....	39
B.4	错误检查包（PEC）实现.....	40
B.5	通信时钟速度和时钟拉伸.....	40
B.6	ALERT、DATA和CLOCK线的上拉电阻配置.....	40
B.7	局限性.....	40
B.8	主机PC到USB接口适配器命令列表.....	41
B.9	USB接口适配器到主机PC响应命令列表.....	42
B.10	发送字节命令和响应.....	43
B.11	接收字节命令和响应.....	43
B.12	编写字节命令和响应.....	43
B.13	编写Word命令和响应.....	44
B.14	读取字节命令和响应.....	44
B.15	阅读Word命令和响应.....	45
B.16	进程调用命令和响应.....	45
B.17	块写入命令和响应.....	46
B.18	块读取命令和响应.....	46
B.19	块读取-块写入过程调用命令和响应.....	47
B.20	群组命令和响应.....	47
B.21	控制线命令与响应的断言/取消断言.....	48
B.22	PMBUS信号线命令与响应的Poll.....	48
B.23	打开/关闭PEC命令和响应.....	49

图列表

1-1 USB接口适配器、USB电缆和带状电缆的连接.....	9
4-1 硬件编程确定流程图.....	22
4-2 固件编程期间应用程序的交互.....	23
5-1 USB适配器软件架构.....	25
5-2 样本GUI应用程序.....	28
6-1 示意图.....	30
6-2 PCB布局——顶部L1（铜层）——装配TA.....	33
6-3 PCB布局——顶部S1（丝网印刷层）.....	34
6-4 PCB布局——顶部L1（铜层）.....	35
6-5 PCB布局——底部L2（铜层）.....	36

表列表

2-1 终端功能.....	11
2-2 引脚8、9和的上拉电阻器的可配置选项 10.....	12
4-1 固件版本所需的命令对列表.....	20
4-2 固件版本的三字编码.....	20
4-3 初始家庭代码.....	21
6-1 材料清单.....	31
B-1 主机PC到USB接口适配器命令.....	41
B-2 USB接口适配器到主机PC的响应命令.....	42

请先阅读

关于本手册

本用户指南从硬件设计、固件编程、通信协议、GUI和PC库等方面描述了USB接口适配器评估模块的功能和操作。

如何使用本手册

本文件包含以下章节：

- 第1章简介
- 第二章硬件设计概述
- 第三章固件编程与通信协议
- 第四章固件版本方案
- 第五章GUI和PC库
- 第6章原理图、物料清单和PCB布局

注意事项和警告信息



CAUTION

本EVM包含可能因静电放电而损坏的组件。在不使用时，请始终将EVM用其提供的防静电袋进行运输和储存。操作时请使用防静电腕带，并在防静电工作台上操作。有关正确操作的更多信息，请参阅SSYA008。

WARNING

By default, the I²C data and clock lines are pulled up internally to 3.3 V. If internal pullup resistors are not used and external ones are used instead, please make sure to pull up to 3.3 V only. Operation from 5 V is not specified and may permanently damage this USB Interface Adapter EVM.

警示或警告中的信息是为了保护您而提供的。请仔细阅读每个警示和警告。

FCC 警告

面心立方警告

此设备仅适用于实验室测试环境。它能够生成、使用并可能辐射射频能量，但尚未根据面心立方规则第15部分J子部分的规定进行测试，该规定旨在提供合理的射频干扰防护。如果在其他环境中使用此设备，可能会对无线电通信造成干扰，用户需自行承担费用，采取必要措施解决此类干扰问题。

这个EVM作为接口适配器或桥接器，连接主机PC（兼容IBM™）与一个或多个从设备，通过标准A型到mini-B型的USB线缆。USB接口适配器与主机PC之间的通信通过USB实现，而与从设备之间的通信则通过集成电路（I²C）和/或通用输入/输出（GPIO）实现。该桥接器负责在USB和I²C/GPIO之间转换通信事务。

注意：为了使USB接口适配器与多个I²C从设备通信，需要制作一根带有多个连接器的特殊10针扁平电缆，这样所有I²C从设备就可以在同一条扁平电缆上串联起来。

1.1 操作用硬件项目清单

为了操作这个USB接口适配器，需要以下物品（见图1-1）：

- 机箱中的USB接口适配器
- 标准A型至mini-B型（5针）USB连接线
- 两端带有连接器的10针带状电缆



图1-1。USB接口适配器、USB电缆和带状电缆的连接

1.2 操作用软件项目列表

除前面列出的硬件项目外，还需要以下软件：

- 由德州仪器 (TI) 开发的I²C/GPIO/PMBUS/SMBUS传输层DLL驱动程序(PMBus传输(USB HID) .dll)
- 适用于I²C/GPIO/PMBUS/SMBUS应用的USB SAA GUI.exe演示GUI软件

两者均可从TI网站的文件“USB接口适配器GUI.zip”下载，该文件还包含其他附件文件。

PC将USB接口适配器识别为通用人机接口设备 (HID)，Windows操作系统内置的USB/HID驱动程序支持该设备。因此，USB接口适配器是即插即用的，不需要专有的USB驱动程序。

1.3 操作和故障排除的分步说明

下载并安装传输DLL和GUI演示软件后，请按照以下步骤操作USB接口适配器：

1. 将USB电缆插入PC和USB接口适配器，等待绿色LED灯亮起。
故障排除：如果30秒后绿色LED未点亮，请检查以确保USB电缆已牢固连接。如果连接是安全的，请尝试使用其他USB端口。如果使用其他USB端口不能解决问题，请尝试重新启动计算机。如果重新启动计算机不能解决问题，请尝试更换USB电缆。如果更换USB电缆不能解决问题，请联系TI技术支持以获取帮助。
2. 将10针扁平电缆同时插入USB接口适配器和I²C从板。确保扁平电缆连接器上的凹槽与机箱插座的孔位相匹配。
3. 启动I²C从板。请注意，默认的I²C速度设定为100 kHz，而I²C数据线和时钟线的默认上拉电阻值为2.2 kΩ。
4. 运行DEMO GUI软件并按照GUI的说明进行操作。
故障排除：如果I²C从板无法与USB接口适配器通信，首先检查10针扁平电缆是否已正确连接。如果电缆已正确连接，检查I²C通信速度，并确保I²C数据线和时钟线的上拉电阻适合该应用。如果还是不能解决，请修改I²C通信速度和上拉电阻。如果还是不能解决，请联系I²C从板的制造商寻求帮助。

本章介绍USB接口适配器硬件设计的主要特点。有关原理图、物料清单（BOM）和PCB布局信息，请参见第6章。

2.1 介绍

硬件基于德州仪器（TI）的USB外围芯片（TUSB3210）。该芯片搭载了性能增强的8052核心。电路板采用简单的双层设计，顶部布线（详见第6章的原理图）。硬件设计的主要特点将在后续章节中详细阐述。

2.2 带状电缆连接器J2和引脚排列

此连接器用于USB接口适配器和一个或多个I²C从设备之间的通信和控制。

表2-1终端功能

航站楼		I/O	描述
姓名	不		
PMBCTRL5/GPIO7	1	I/O	用作第5个PMBUS控制线（输出）或GPIO引脚7（输入/输出，启用内部上拉）
PMBCTRL4/GPIO6	2	I/O	用作第4个PMBUS控制线（输出）或GPIO引脚6（输入/输出，启用内部上拉）
PMBCTRL3/GPIO5	3	I/O	用作第3个PMBUS控制线（输出）或GPIO引脚5（输入/输出，启用内部上拉）
PMBCTRL2/GPIO4	4	I/O	用作第2个PMBUS控制线（输出）或GPIO引脚4（输入/输出，启用内部上拉）
+3.3V	5	功率	此引脚可提供高达100 mA的3.3 V输出电源。只要总电流消耗小于100 mA，任何从属设备(s)都可以使用此电源。
地面	6	地线	整个评估委员会的共同点
PMBCTRL1/GPIO3	7	I/O	用作第1个PMBUS控制线（输出）或GPIO引脚3（输入/输出，启用内部上拉）
PMBALERT/SMBALERT/GPIO2	8	I/O	用作PMBUS或SMBUS通信的ALERT线（输入），或用作GPIO引脚2（输入/输出，禁用内部上拉，启用外部可编程上拉）
PMBC/SMBC/SCL/GPIO1	9	I/O	用作PMBUS、SMBUS或I ² C通信的时钟线（输出），或用作GPIO引脚1（输入/输出，禁用内部上拉，启用外部可编程上拉）
PMBD/SMBD/SDA/GPIO0	10	I/O	用作PMBUS或SMBUS或I ² C通信的数据线（输入/输出），或用作GPIO引脚0（输入/输出，禁用内部上拉，启用外部可编程上拉）

2.3 引脚8、9和10的上拉电阻配置

引脚8、9和10的上拉电阻（参见第6章的电路图）可以通过与TUSB3210中的嵌入式固件通信进行配置（参见第3章的命令“设置上拉电阻”）。默认情况下，这三个引脚的上拉电阻均为2.2 k Ω 。表2-2列出了每个引脚的可能选项。

表2-2.引脚8、9和10的上拉电阻器可配置选项

拉力电阻器 选项 (字节值)	选项1 (0x00)	备选案文2 (0x01) (默认)	选项3 (0x02)	选项4 (0x03)
J2的第8引脚：报警线	无拉起 (打开)	2.2 k Ω	N/A	N/A
J2的第9引脚：时钟线	无拉起 (打开)	2.2 k Ω	1 k Ω	688 Ω
J2的第10引脚：数据线	无拉起 (打开)	2.2 k Ω	1 k Ω	688 Ω

2.4 电可擦除只读存储器

嵌入式固件存储在64Kbit的EEPROM中（参见第6章原理图中的U1）。固件支持现场编程（更多详情请参见第5章）。跳线分流器J1为常闭型，仅用于固件开发。上电后，TUSB3210从EEPROM下载固件映像到内部静态RAM并开始执行固件（首先进行USB枚举）
从那里

2.5 USB连接的LED指示

评估板USB接口旁边装有一个绿色LED指示灯，每次USB接口适配器通过USB线连接到PC机上，主机PC机检测到它并成功完成USB枚举后，嵌入式固件点亮绿色LED（D1）指示；
否则，LED保持熄灭。拔掉USB线后，LED会立即熄灭。

固件编程和通信协议

以下部分描述了嵌入式固件，以及USB接口适配器支持的通信协议和如何使用协议中的每个命令。

3.1 固件功能摘要

TUSB3210中的嵌入式固件支持以下主要功能：

- 支持USB1.1全速模式，传输速率可达12 Mbit/s。USB接口适配器被主机识别为通用人类接口设备（HID），因此无需安装专有的USB或HID驱动程序。因此，该USB接口适配器可视为即插即用的USB设备。
- 支持在外部EEPROM中进行固件编程，以便在现场更新固件。有关固件编程的[详细信息](#)，请参见TI网站（www.ti.com）。
- 为ALERT、DATA和CLOCK线配置不同的上拉电阻
- 支持灵活的I²C读写事务。USB接口适配器用作I²C主设备。
- 为I²C、PMBUS或SMBUS通信选择不同的时钟速度（100 kHz或400 kHz）
- 提供固件版本信息（有关固件版本方案的详细信息，请参见第4章）

3.2 固件版本命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB 数据包 字 节数6至63	评论
固件版本（获取 版本）	0x00							见第4章

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB 数据包 字 节数6至63	评论
固件版本的响应	0x00							见第4章

有关如何分配和解释“家庭代码”、“主要版本”和“次要版本”字节的详细说明，请参阅第4章。

I²C写入命令和响应

3.3 I²C写入命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
C型I ² 写入	0x14	A	B	C	要写入的C字节数（从1到60字节）			

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
I ² C写入响应	0x94	成功: 0 失败: 1						

以下详细描述了此命令的位操作：I²C开始→写入A（即从设备地址左移一位+写入0）→从设备确认→写入B（通常是命令字节或寄存器地址字节）→从设备确认→重复C次（每次写入1字节→从设备确认）→I²C结束。第3.10节中还介绍了一个更通用的I²C写入事务命令。

3.4 I²C读取命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
I ² C读取	0x15	A	B	C	D（从1到 62字节）			

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
I ² C读取的响应	0x95	成功: 0 失败: 1	读取的D字节数					

以下详细描述了此命令的位操作流程：I²C开始→写入A（即从地址左移一位+写入0）→从设备确认→写入B（通常是命令字节或寄存器地址字节）→从设备确认→重复开始→写入C（即从地址左移一位+读取1）→从设备确认→从从设备读取D字节（期间有主设备确认）→主设备确认→I²C停止。第3.11节中还介绍了一个更通用的I²C读取事务命令。

3.5 读/写端口0命令和响应

命令描述	USB数据包字节0 命令编码	USB数据包字节1	USB数据包字节2	USB数据包字节3	USB数据包字节4	USB数据包字节5	USB数据包字节6至63	评论
读/写端口0	0x16	A (bit.x=1表示 read/input, 当bit.x=0时 写入/output)	B (要写入的字节数)					B不适用于端口0的读取。每个GPIO可以配置为输入或输出。

命令描述	USB数据包字节0 命令编码	USB数据包字节1	USB数据包字节2	USB数据包第3字节	USB数据包字节4	USB数据包字节5	USB数据包字节6至63	评论
响应来自读/写端口0	0x96	成功: 0 失败: 1	字节读取从端口0					

TUSB3210端口0的每个GPIO引脚都可以配置为输入或输出（请参见第6章中的原理图）。

GPIO引脚3到7被配置为具有打开的内部上拉，因此不需要特殊的外部上拉，除非需要更强的I/O驱动。

GPIO引脚0、1和2被配置为开漏I/O端口。使用软件可配置的上拉电阻来支持I/O功能。

当引脚0或引脚1用作GPIO时，I²C功能将无法正常工作。只有在完成上电复位后，才能重新启用I²C功能，此时无需再使用引脚0和1作为GPIO。因此，建议最多使用六个GPIO引脚，并且应将引脚0和1专门用于I²C应用。

3.6 EEPROM程序命令和响应

命令描述	USB数据包字节0 命令编码	USB数据包字节1	USB数据包字节2	USB数据包字节3	USB数据包字节4	USB数据包字节5	USB数据包字节6至63	评论
电可擦除只读存储器程序	0x18	A=起始地址MSB	B=起始向LSB发送	C (#字节) 程序编制	要编程到EEPROM中的数据（最多32字节）			

命令描述	USB数据包字节0 命令编码	USB数据包字节1	USB数据包字节2	USB数据包字节3	USB数据包字节4	USB数据包字节5	USB数据包字节6至63	评论
EEPROM 程序的响应	0x98	成功: 0 失败: 1						

此命令用于将固件映像编程到EVM的EEPROM（原理图上的U1）。建议将整个固件映像分成多个32字节的块，并重复调用此命令，直到完成编程。

3.7 EEPROM读取命令和响应

命令描述	USB数据包字节0 命令编码	USB数据包字节1	USB数据包字节2	USB数据包第3字节	USB数据包字节4	USB数据包字节5	USB数据包字节6至63	评论
EEPROM读取	0x19	A=起始地址MSB	B=起始向LSB发送	C (#字节) 读				C ≤ 60

命令描述	USB数据包字节0 命令编码	USB数据包字节1	USB数据包字节2	USB数据包字节3	USB数据包字节4	USB数据包字节5	USB数据包字节6至63	评论
响应来自EEPROM读取	0x99	成功: 0 失败: 1	从EEPROM读回的C字节数 (最多32个字节)					

此命令与EEPROM程序命令一起使用，以验证固件映像是否已正确编程到EEPROM中。为了达到最佳效率，请尝试通过调用此命令一次读取32字节的块。

3.8 设置上拉电阻器命令和响应

命令描述	USB数据包字节0 命令编码	USB数据包字节1	USB数据包字节2	USB数据包第3字节	USB数据包字节4	USB数据包字节5	USB数据包字节6至63	评论
设置上拉电阻器	0x1A	A (SDA)	B (用于SCL)	C (用于ALE-RT)				0x00 表示无上拉 (已关闭)，0x01 表示 2.2 kΩ, 0x02 表示 1 kΩ, 0x03 表示 688 Ω。对于 ALE-RT 功能，选项为开启 (0x00) 或 2.2 kΩ (0x01)。

命令描述	USB数据包字节0 命令编码	USB数据包字节1	USB数据包字节2	USB数据包第3字节	USB数据包字节4	USB数据包字节5	USB数据包字节6至63	评论
设置上拉电阻器的响应	0x9A	成功: 0 失败: 1						已配置引体向上

字节A用于I²C、PMBUS或SMBUS通信的数据线；字节B用于I²C、PMBUS或SMBUS通信的时钟线；字节C用于PMBUS通信的警报线。如果从设备能够提供一个或多个上拉电阻，应关闭数据线、SCL线或警报线的上拉电阻。

3.9 I²C-PMBUS-SMBUS速度命令与响应设置

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字 节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
设置 I ² C-PMBUSSMBUS速度	0x1 B	A (0, 对 于 100 kHz, 否则 400 kHz)						

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字 节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
来自集的响应 I ² C-PMBUSSMBUS速度	0x9 B							快速完成 挑选

对于I²C、PMBUS或SMBUS通信，时钟速度可设置为100 kHz或400 kHz。作为I²C、PMBUS或SMBUS的主设备，USB接口适配器支持时钟拉伸（每事务最长可达25毫秒）。固件将默认时钟速度设为100 kHz。

3.10 通用I²C写入命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
通用I ² C写入	0x1C	A	要写入的字节数（从2到62）					要写入的数据 的第一个字节 通常是（设备 地址 < 1 + 写入位0）。

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
通用I ² C写入的响 应	0x9C	成功：0 失败：1						

这是一个通用的I²C写事务命令，适用于大多数I²C应用。具体操作步骤如下：首先执行I²C启动→重复→（写入1个数据字节，从设备确认）A次→最后I²C停止。

基本上，所有交易字节依次放置，例如地址字节（7位从设备地址左移1位+写入位为0）、一个或多个命令字节、一个或多个数据字节，这些字节被放入A字节数的缓冲池中，然后USB接口适配器执行位银行操作。为了实现最大的灵活性，建议使用这种通用命令来处理任何I²C写入事务。

3.11 通用I²C读取命令和响应

命令 描述	USB数据包 字节0 命令 编码	统一的S波段 包裹 字节1	统一的S波段 包裹 字节2	统一的S波段 包裹 第3字节	统一的S波段 包裹 字节4	统一的S波段 包裹 字节5	USB数据包 字节6至63	评论
通用I ² C读取	0x1D	A	要写入的字节数 (2至60)				Y = 1st 字节转为 写+ 读取位1 统一的S波 段 字节62 X#的 字节 (1 至62)至 复述 统一的S波 段 字节63	要写入的数据 的第一个字节 通常是 (设备 地址 < 1 + 写入位0)。

命令 描述	USB数据包 字节0 命令 编码	统一的S波段 包裹 字节1	统一的S波段 包裹 字节2	统一的S波段 包裹 第3字节	统一的S波段 包裹 字节4	统一的S波段 包裹 字节5	USB数据包 字节6至63	评论
通用I ² C读取的 响应	0x9D	成功: 0 失败: 1	从从机读回的X个字节 (1到62)					

这是一个通用的I²C读取事务命令，可以应用于大多数I²C应用程序。详细的位银行操作如下：I²C开始→重复写入1个数据字节并接收从设备的确认→重复此步骤A次→写入Y个字节→从从设备读取X个字节（期间主设备发送确认）→主设备发送NACK响应→I²C结束。

基本上，所有交易字节依次排列，例如地址字节（7位从设备地址左移1位+写入位为0）、一个或多个命令字节，然后重复开始、写入地址字节Y（7位从设备地址左移1位+读取位为1），并从被寻址的从设备读回x个字节。USB接口适配器负责位银行操作。为了实现最大的灵活性，建议使用这种通用命令来执行任何I²C读取事务。

3.12 板测试命令和响应

命令 描述	统一的S波段 包裹 字节0 命令 编码	统一的S波段 包裹 字节1	统一的S波段 包裹 字节2	统一的S波段 包裹 第3字节	统一的S波段 包裹 字节4	统一的S波段 包裹 字节5	统一的S波段 包裹 字节6至63	评论
形板测验	0x7F							开始同时切换连接到八个GPIO引脚的八个LED，持续5秒。

命令 描述	统一的S波段 包裹 字节0 命令 编码	统一的S波段 包裹 字节1	统一的S波段 包裹 字节2	统一的S波段 包裹 字节3	统一的S波段 包裹 字节4	统一的S波段 包裹 字节5	统一的S波段 包裹 字节6至63	评论
董事会测试的 响应	0xFF							完成板测试

此命令仅用于生产期间对USB接口适配器评估板进行内部测试。请勿将此命令用于任何其他用途。

3.13 其他用于不同应用程序的命令

对于I²C/GPIO以外的应用，如PMBUS、SMBUS、HDQ等，请参阅相关文献以了解详细信息。

固件版本方案

本章简要介绍了USB接口适配器的固件版本控制方案。了解此方案对于在预配置中使用设备至关重要。对于希望定制实现固件或了解主机客户端软件如何验证固件版本的用户来说，本主题非常重要。

4.1 介绍

固件的大小不得超过8K字节。这限制了单个固件映像的功能数量。每个应用所需的所有功能不可能全部包含在这个空间内。为了使设备具有可扩展性，并支持多种应用，设计时允许主机应用程序在需要时使用自定义固件重新编程设备。

例如，某些终端设备可能需要协议X来发送和接收数据，但标准固件不支持协议X。在这种情况下，支持此终端设备的GUI软件的程序员将编写符合第4.2节规则的自定义固件。在运行时，GUI软件会查找USB接口适配器并检索其固件版本号。如果版本号与所需的协议不兼容，则GUI软件会使用自己的固件版本重新编程设备，以便其可以使用该版本。

如果所有GUI软件都符合此版本系统，则相同的硬件可以重复用于许多应用程序，而不会妨碍其他符合的软件。这也确保了所有的实现与较旧的GUI软件向后兼容。

4.2 实现合规自定义固件的规则

为了使自定义固件与这个版本方案兼容，它必须遵守以下规则：

1. 自定义固件必须支持表4-1中列出的命令对（默认实现请参见第3章）。

表4-1：固件版本所需的命令对列表

代码（十六进制）	描述	信息
0x00和0x80	固件版本（获取版本）	需要向GUI软件传达其独特身份
0x18和0x98	EEPROM专业版	GUI软件需要重新编程嵌入式EEPROM上的固件
0x19和0x99	EEPROM读取	GUI软件需要验证EEPROM上的固件映像

2. 自定义固件必须具有唯一且符合要求的版本号，该版本号描述了其功能。

注：这两条规则共同描述了所有GUI软件可以预期实现的接口。如果GUI软件不能调用固件版本命令，它就不能识别固件，也不应尝试使用它，因为结果是不可预测的。如果GUI软件需要重新编程USB接口适配器，它期望命令0x18是EEPROM编程。符合该标准可确保固件不会破坏验证和重新编程循环。

4.3 符合要求的版本控制

版本方案使用三个字节组成一个唯一标识符，设备可以作为数据部分返回该标识符以响应固件版本命令（更多详情请参见第3章）。

第一个字节是家族代码，用于标识固件的责任方。这些代码以块的形式分发给不同的团队和TI。例如，数字电源组拥有代码01至16（或十六进制0x01至0x10）。一些家族代码被预留。家族代码240至255（或十六进制0xF0至0xFF）被留作终端用户或任何希望制作符合标准的自定义固件版本以供个人使用的人使用。

此设备的“设备代码”还表示支持的协议的排列方式，例如，“设备代码”1支持I²C、PMBus和SMBus，而“设备代码”2可能支持I²C和协议X。

第二个字节是主要版本，它描述了协议排列的修订。如果支持的协议需要新功能或对调用方式（签名更改）进行了更改，则此字节表示变更。

第三个字节是次要版本，它描述了协议排列的修订。如果固件代码已更改以修复功能中的错误，但没有添加新的公共功能，也没有对函数签名进行任何更改，则应更改此字节以表示新版本。

表4-2总结了三字节代码的意义。

表4-2.固件版本的三个字节

字节顺序	最显著字节	中间字节	最低有效字节
字节含义	家庭法	主要版本	次要版本
示例值	01 = 由支持Digital Power PMBus、SMBus和I ² C的设备所拥有	01 = 协议排列的重大修订	01 = 协议排列的轻微修订

为了正确实施这一版本控制方案，Windows™ 应用程序会调用一个固件版本命令，该命令与其他一些命令一起被预留，并且必须在每个设备的实现中存在。如果家族代码与应用程序查找的代码不匹配，应用程序不应假设支持任何协议。家族代码应立即尝试使用其自身版本（随应用程序分发的二进制映像）重新编程固件。

理论上，一个将家族代码和主要版本与固件实现关联的Windows应用程序，只需确保次要版本不低于目标次要版本。固件程序员应谨慎处理，避免从具有相同家族代码和主要版本的实现中移除功能，否则旧版Windows应用程序在尝试通信时可能会发出错误调用。如果确实需要进行重大更改，建议固件开发者修改家族代码字节。

表4-3.初始家庭代码

家庭代码 (十进制)	组/团队
00	已预订
01–16	数字电源+ POE
17–32	HPA设计工具团队
33–48	电池管理
49–239	未分配
240–255	开放以进行自定义实现 (用户定义)

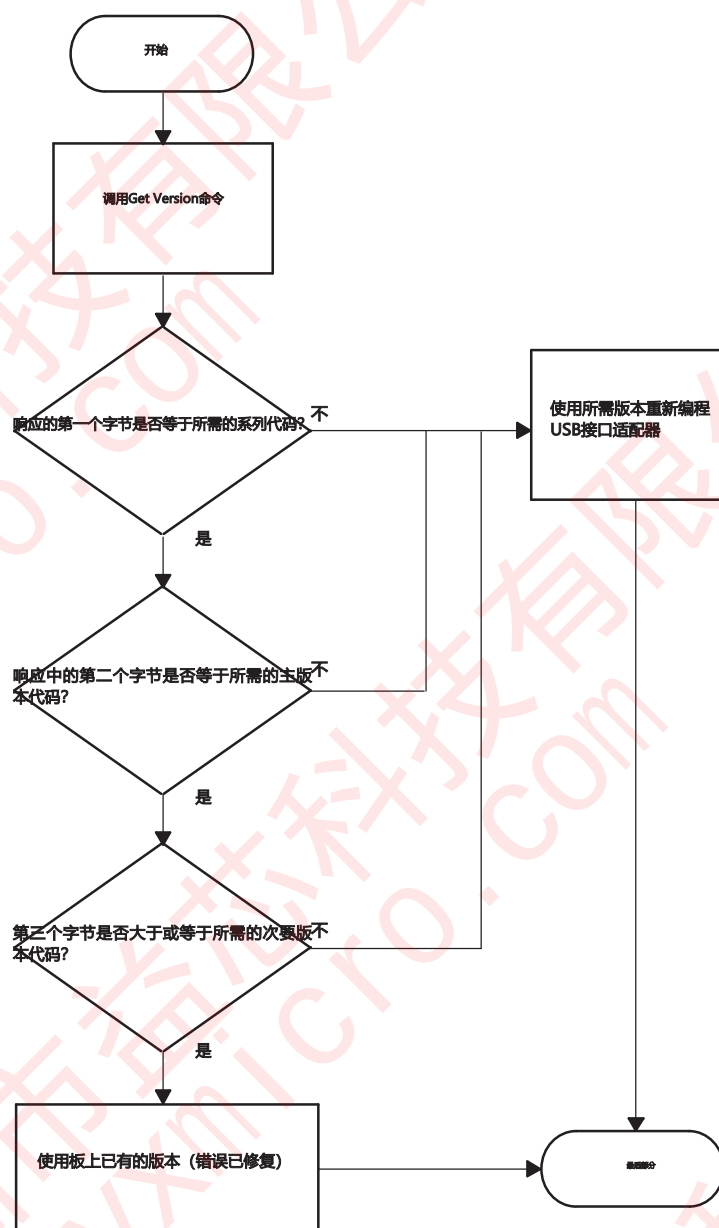


图4-1：固件编程确定流程图

图4-2显示了固件编程到EEPROM之间如何工作的示例
申请

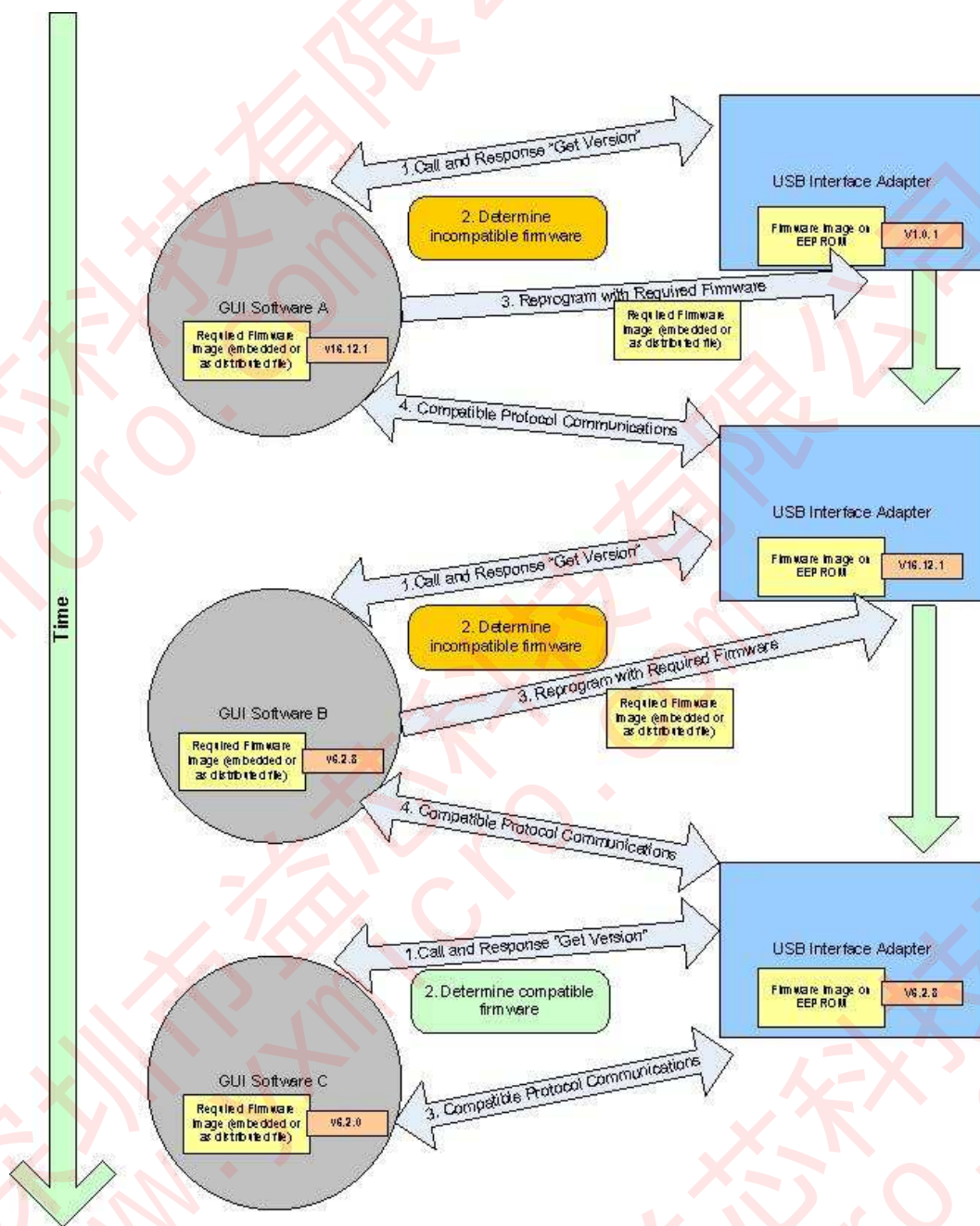


图4-2：固件编程期间应用程序的交互

适配器解决方案包括一组PC库和GUI，可用于允许PC应用程序访问适配器和支持的适配器目标设备。

GUI和PC库是使用Microsoft®.NET Framework设计的，旨在充分利用标准库。如本文档所述，适配器以USB HID的形式向Windows提供服务。因此，无需自定义驱动程序，PC上的应用软件可以使用标准的Microsoft Windows USB驱动程序（usbhid.sys）。

与适配器一样，Windows库支持所有类型的设备接口：I²C、SMBus、PMBus和GPIO。此外，这些库还支持通过允许PC程序更新适配器中的EEPROM、选择在数据线和时钟线上切换电阻等操作来配置和控制适配器。

图5-1表示所用PC的软件架构。

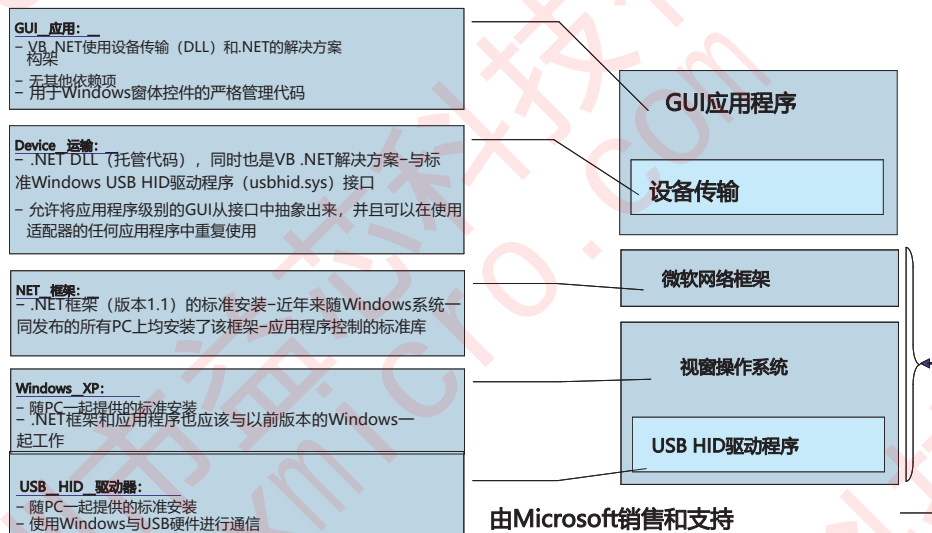


图5-1 USB适配器软件架构

软件体系结构的主要部分如下：

- Windows XP™
- 微软网络框架
- 器械运输
- GUI应用程序

5.1 视窗操作系统

Windows XP通过随Microsoft Windows提供的标准USB HID驱动程序，为应用程序库提供标准的USB HID支持。

.NET Framework是微软提供的一套库，它允许Windows应用程序使用一系列丰富的服务。.NET Framework支持表单和窗口管理、内存管理和数据库等功能。大多数最近为Windows编写的应用程序都使用了.NET Framework。

.NET框架是设备传输库和GUI应用程序的唯一依赖项；此应用程序不需要任何自定义驱动程序。.NET框架已安装在所有新PC上，因此除了解压缩并运行适配器框架提供的GUI应用程序外，无需执行任何操作。

当前版本的设备传输库和GUI应用程序使用的是Microsoft .NET框架的1.1版。根据生成GUI库和应用程序所使用的开发工具的版本，可能需要更新Microsoft的.NET框架。

要更新Microsoft的URL。您可以在以下URL中找到您的PC上的.NET Framework：

<http://www.microsoft.com/downloads/details.aspx?FamilyID=0856EACB-4362-4B0D-8EDD-AAB15C5E04F5&displaylang=en>

5.2 设备传输

为了使适配器的用户能够以最小的努力创建新的应用程序，提供了一个设备传输库。

此库是一个使用Microsoft Developer Studio®2003生成的.NET DLL，基本上是一组.NET管理代码。此库要求在使用它的计算机上安装.NET Framework 1.1版。

该库包含在适配器提供的示例GUI应用程序中，旨在由其他应用程序使用，并通过适配器管理目标。

要使用此库，只需将其作为依赖项包含在正在开发的项目中。

此库提供以下功能：

- 支持适配器所有I/O类型的原语
 - I²C
 - I²C读取（可变长度）
 - I²C写入（可变长度）
 - SMBus
 - 接收字节
 - 读取字节
 - 阅读单词
 - 屏蔽阅读
 - 发送字节
 - 写入字节
 - 写单词
 - 区块写入
 - 进程调用
 - 块写入、块读取及进程调用
 - PEC enable/disable
 - PMBus
 - 群组命令
 - 控制线的声明/断言
 - 警报#信号
 - GPIO（读/写）
- 一般功能
 - 重置适配器
 - 获取设备版本
 - 选择内部上拉电阻器
 - 更新/读取EEPROM内容
 - 选择100 kHz/400 kHz总线速度

5.3 GUI应用程序

适配器附带的示例应用程序使用设备传输与适配器以及适配器后面的靶标设备进行通信。图5-2中显示了屏幕截图。

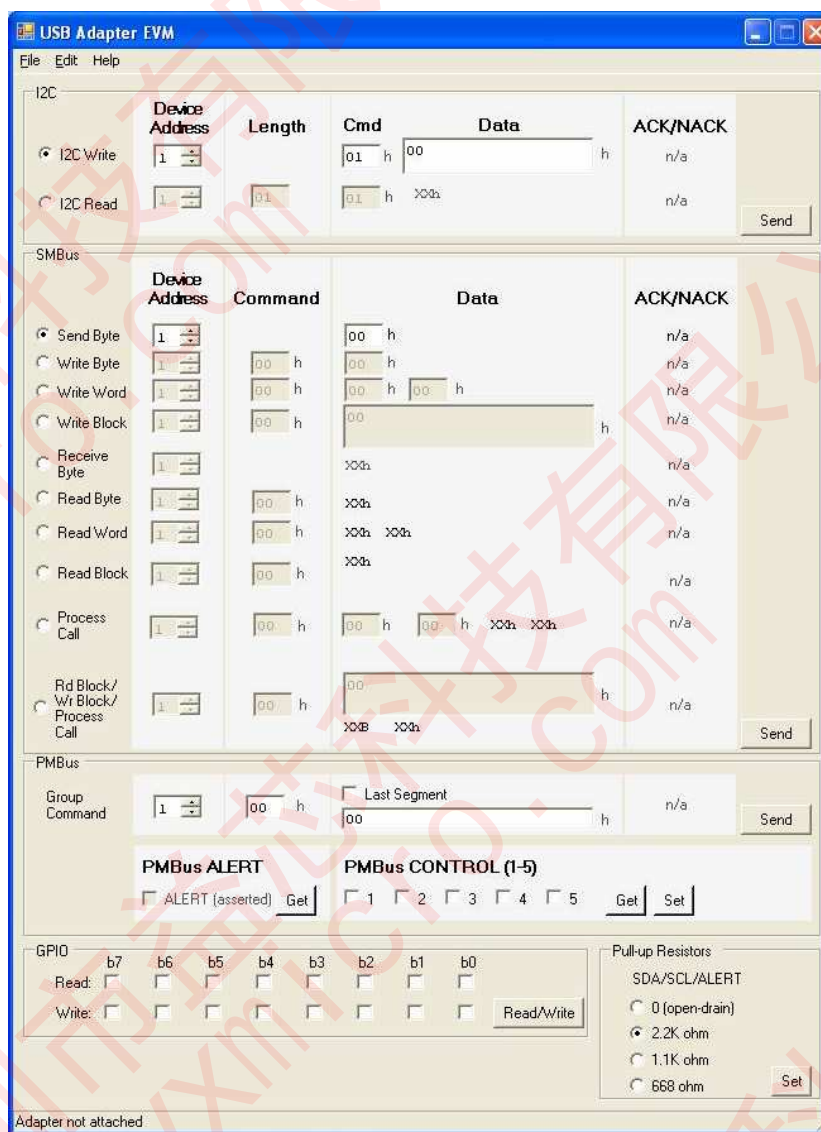


图5-2.示例GUI应用程序

从主菜单中，更新适配器的固件，选择设备的总线速度（100 kHz或400 kHz），并选择在与设备通信时启用或禁用PEC。

表单的主要部分分为五个部分：I²C、SMBus、PMBus、GPIO和上拉电阻。用户可以从I²C、SMBus和PMBus组框中选择适配器支持的每种事务类型。此外，设备的事务结果也会在此处以确认（ACK）或未确认（NACK）的形式传达。

在GPIO组框中，八个GPIO引脚中的每一个都可以被启用，并且可以被编程为读取或写入。此外，还可以从这个组框中读取引脚的状态，或者将其设置为所需的值。

最后，GUI允许用户选择用于SDA/SCL线的上拉电阻器的值。适配器和GUI允许用户从668 Ω、1.1 kΩ、2.2 kΩ或开漏中进行选择。

示意图、物料清单、PCB布局

本章包含有关原理图设计的信息，以及有关USB接口适配器的物料清单（BOM）和印刷电路板（PCB）布局的相关信息。

6.2 物料清单 (BOM)

表6-1: 材料清单

量	回复 指定者	价值	描述	尺寸	部件编号	制造商
6	C1, C5, C6, C8, C10, C11	0.1 μ F	陶瓷电容器, 0.1 μ F, 25 V, X7R, 10%	0603	C1608X7R1E104KT	哈萨克斯坦
2	C2, C3	22 pF	电容, 陶瓷, 22皮法, 50伏, C0G, 10%	0603	C1608C0G1H220KT	哈萨克斯坦
1	C4	1000 pF	电容, 陶瓷, 1000皮法, 100伏, C0G, 5%	0805	标准	标准
1	C7	1 μ F	钽电容器, 1 μ F, 16 V, 20%	3216	293D105X0016A2T	维夏
2	C9, C12	10 μ F	钽电容器, 10 μ F, 10 V, 20%	3216	293D106X0010A2T	维夏
1	D1	SSF-LXH305GD-TR	二极管, LED, 2.6 V, 25 mA	0.25 $\times$ 0.25	SSF-LXH305GD-TR	卢美克斯
1	D2	MBRA130型	肖特基二极管, 1A, 30 V	SMA	MBRA130型	红外
1	D3	7.5 V	二极管, 稳压器, 7.5 V, 3 W	中小企业	1SMB5922BT3	论半
8	D4, D5, D6, D7, D8, D9, D10, D11	GL05T型	二极管, TVS二极管, 低电容	SOT23封装	GL05T型	将军
1	J1	PTC36SAAN	头, 2针, 间距100 mil (36针条)	0.1 $\times$ 2	PTC36SAAN	苏林斯
1	J2	86479-3	连接器, 公端直角2 $\times$ 5针, 间距100 mil, 4 Wall	0.607 $\times$ 0.484	86479-3	安培
1	J3	UX60-MB-5ST	连接器, 接收端口, USB-B, 迷你型, 五针, SMT	0.354 $\times$ 0.303	UX60-MB-5ST	希罗斯
1	Q1	MMBT2222A	NPN型晶体管, 高性能, 500 mA	SOT-23	MMBT2222A	费尔柴尔德绰号
5	Q2, Q3, Q4, Q5, Q6	BSS84	MOSFET, Pch 型, -50V, -0.13 A, 10 Ω	SOT-23	BSS84	费尔柴尔德绰号
3	R1, R2, R25	1.5 k Ω	电阻器, 芯片, 1.5 k Ω , 1/16 W, 5%	0603	标准	标准
3	R17, R19, R31	2.2 k Ω	电阻器, 芯片, 2.2 k Ω , 1/16 W, 5%	0603	标准	标准
2	R18, R20	1 k Ω	电阻器, 芯片, 1 k Ω , 1/16 W, 5%	0603	标准	标准
1	R29	200 Ω	电阻器, 芯片, 200 Ω , 1/16 W, 5%	0603	标准	标准
1	R3	1 M Ω	电阻器, 芯片, 1 M Ω , 1/16 W, 1%	0603	标准	标准
2	R4, R28	15 k Ω	电阻器, 芯片, 15 k Ω , 1/16 W, 5%	0603	标准	标准
5	R5, R14, R15, R16, R30	100 k Ω	电阻器, 芯片, 100 k Ω , 1/16 W, 5%	0603	标准	标准
14	R6, R7, R8, R9, R10, R11, R12, R13, R21, R22, R23, R24, R26, R27, R32	33 Ω	电阻器, 芯片, 33 Ω , 1/16 W, 5%	0603	标准	标准
1	U1	24LC64	IC, 串行EEPROM 64 k Ω , 1.8 V至5.5 V, 最大400 kHz	索8	24LC64	微芯片
1	U2	TUSB3210PM	IC、USB、通用设备控制器	0.48 $\times$ 0.48	TUSB3210PM	钛
1	U3	TPS76333DBV	IC, 微功率型150 mA LDO稳压器	SOT23-5	TPS76xxxDBV型	钛
1	Y1	12 兆赫		0.185 $\times$ 0.532	CY12BPSMD	克瑞斯特克
1		M3DDA-1018J-ND	带状电缆, 插座对插座, 10针	18	M3DDA-1018J-ND	3M
1			分流器, 100 mil, 黑色	0.1	929950-00	3M
1			PCB, 3.12英寸 $\times$ 1.6英寸 $\times$ 0.062英寸		HPA172E2	任意

物料清单 (BOM)

注：这些组件对ESD敏感，因此应遵守ESD预防措施。

这些组件必须清洁，无助焊剂和所有污染物。使用无助焊剂是不可接受的。

这些组件必须符合IPC-A-610第2类工艺标准。

带有星号（“*”）的参考编号不能被替换。所有其他组件都可以用制造商的等效组件替换。

6.3 印刷电路板 (PCB) 布局

图6-2至图6-5显示了PCB布局。

TEXAS INSTRUMENTS		Copper Layer Name		Silkscreen		S Mask		P Mask		Assembly		Fab Drawing
Board No.	HPA172	Top	Bot	Top	Bot	Top	Bot	Top	Bot	Top	Bot	
Rev.	E2	L1								TA		
Date: 01/03/06	Filename: HPA172A.pcb	Engineer: W. Wang	PCB Dsgnr: S. McGee	Modified Date: Thu May 11, 2006				Time Stamp: 13:56:49				

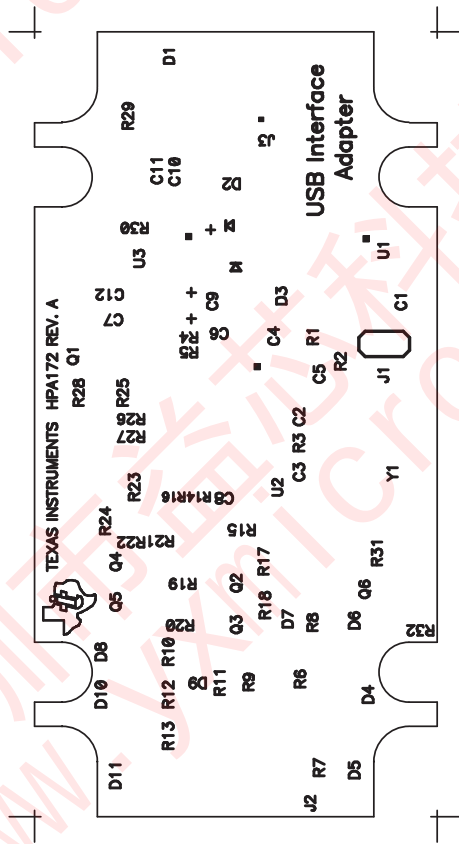
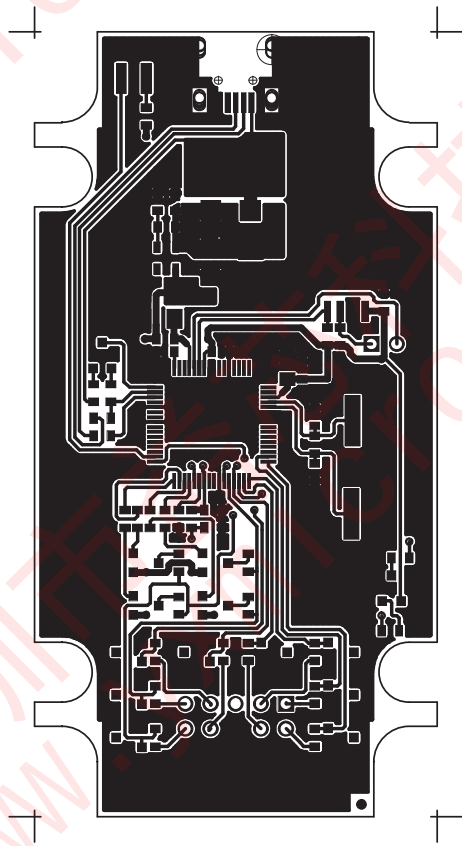


图6-3. PCB布局-Top S1（丝网印刷层）

TEXAS INSTRUMENTS													
Board No.	HPA172	Rev.	E2	Copper Layer Name		Silkscreen		S Mask		P Mask		Assembly	
				Top	Bot	Top	Bot	Top	Bot	Top	Bot	Top	Bot
						S1							
Date: 01/03/06		Filename: HPA172A.pcb		Engineer: W. Wang		PCB Design: S. McGee		Modified Date: Thu May 11, 2006		Time Stamp: 13:56:50			



TEXAS INSTRUMENTS		Copper Layer Name		Silkscreen		S Mask		P Mask		Assembly		Fab Drawing	
Board No. HPA172	Rev. E2	Top	Bot	Top	Bot	Top	Bot	Top	Bot	Top	Bot	Top	Bot
		L1											
Date: 01/03/06	Filename: HPA172A.pcb	Engineer: W. Wang		Modified Date: Thu May 11, 2006		PCB Dgnr: S. McGee		Time Stamp: 13:56:50					

图6-4 PCB布局-Top L1 (铜层)

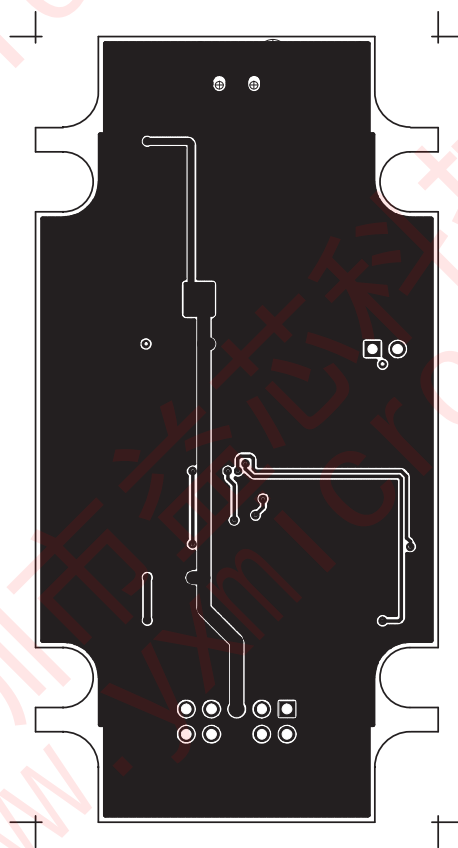


图6-5. PCB布局-底层L2 (铜层)

TEXAS INSTRUMENTS		Copper Layer Name		Silkscreen		S Mask		P Mask		Assembly		Fab Drawing
Board No.	Rev.	Top	Bot	Top	Bot	Top	Bot	Top	Bot	Top	Bot	
HPA172	E2		L2									
Date: 01/03/06	Filename: HPA172A.pcb	Engineer: W. Wang	PCB Dsgnr: S. McGee	Modified Date: Thu May 11, 2006				Time Stamp: 13:56:50				

通信协议列表

本附录包含从主机PC发送到USB接口适配器的通信命令列表，以及从USB接口适配器发送到主机PC的通信命令列表。

A.1 主机PC到USB接口适配器命令

命令 描述	统一的S波段 包裹 字节0 命令 编码	统一的S波段 包裹 字节1	统一的S波段 包裹 字节2	统一的S波段 包裹 字节3	统一的S波段 包裹 字节4	统一的S波段 包裹 字节5	USB数据包 字节6至63	评论
固件版本	0x00							见第4章
I ² C写入	0x14	A	B	C	要写入的C字节数（从1到60字节）			
I ² C读取	0x15	A	B	C	D（从1到 62字节）			
读/写端口0	0x16	A（位x=1，用于 read/input， 当bit.x=0时 write/output）	B（字节到 写）					B不适用于端口0 的读取。每个 GPIO 可以配置 为输入或输出。
电可擦除只读存储器 程序	0x18	A=起始 地址MSB	B=起始 向LSB发送	C（#字节） 程序编制	数据将被编程到EEPROM中 （最多32字节）			
EEPROM读取	0x19	A=起始 地址MSB	B=起始 向LSB发送	C（#字节） 程序编制				C ≤ 60
设置上拉 电阻器	0x1A	A（SDA）	B（用于SCL）	C（用于 警报）				0x00表示无引线 向上（开关 对于OFF，0x01 表示2.2 kΩ；0x 02表示1 kΩ；0 x03 表示 688 Ω。在 ALERT 模 式下，可选参数 为： 打开（0x00）或 2.2 kΩ（0x 01）。
设置 I ² C/PMBUS/SM 总线速度	0x1B	A（0，对于 100 kHz， 否则 400 kHz）						
通用I ² C写入	0x1C	A	要写入的字节数（从2到62）					的第一个字节 要写入的数据是 通常（设备 地址 < 1）+ 写 入位0。
通用I ² C读取	0x1D	A	要写入的字节数（2到60）			Y = 1st 字节转为 写入+ 读取位1 统一的S波 段 字节62	X#的 字节 （1至 62） 读 向后 统一的S波 段	的第一个字节 要写入的数据是 通常（设备 地址 < 1）+ 写 入位0。

					字节63	
--	--	--	--	--	------	--

USB接口适配器向主机PC发送响应命令

命令 描述	统一的S波段 包裹 字节0 命令 编码	统一的S波段 包裹 字节1	统一的S波段 包裹 字节2	统一的S波段 包裹 第3字节	统一的S波段 包裹 字节4	统一的S波段 包裹 字节5	USB数据包 字节6至63	评论
形板测验	0x1F							开始同时切换连接到八个 GP-IO 引脚的八个 LED，持续5秒。

注意：有关每个命令的详细说明，请参见第3章。

A.2 USB接口适配器对主机PC响应命令

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
固件版本的响应	0x80	家庭 数字 (0至255)	主版本 (0至255)	次要版本 (0至255)				
I ² C写入的响应	0x94	成功: 0 失败: 1						
I ² C读取响应	0x95	成功: 0 失败: 1	读取的D字节数					
响应来自 读/写端口0	0x96	成功: 0 失败: 1	字节读取 从端口0					
EEPROM 程序的 响应	0x98	成功: 0 失败: 1						
响应来自 EEPROM读取	0x99	成功: 0 失败: 1	从EEPROM读回的C字节数 (最多32个字节)					
设置上拉电阻器 的响应	0x9A	成功: 0 失败: 1						引体向上已经安排好了。
来自集的响应 I ² C/PMBUS/SM 总线速度	0x9B							快速完成 挑选
通用I ² C写入的响 应	0x9C	成功: 0 失败: 1						
通用I ² C读取的响 应	0x9D	成功: 0 失败: 1	从从机读回X个字节 (1至62)					
董事会测试的响 应	0xFF							完成板测试。

注意：有关每个命令的详细说明，请参见第3章。

PMBUS/SMBUS通信

本附录描述了USB接口适配器对USB到PMBUS/SMBUS通信的支持以及如何在应用程序中使用它。

B.1 概述

USB接口适配器的默认固件支持USB到PMBus (修订版1.0)和SMBus (修订版2.0)的通信。与I²C应用类似, 该适配器作为PMBUS/SMBUS的主设备, 能够通过10针带状电缆与一个或多个PMBUS/SMBUS从设备进行通信。

B.2 支持的基本SMBUS事务类型

由于PMBUS通信是从SMBUS通信派生的, 因此它们共享基本的SMBUS事务类型。这些基本的SMBUS事务类型包括:

- 发送字节
- 接收字节
- 写入字节
- 读取字节
- 写单词
- 阅读单词
- 区块写入
- 屏蔽阅读
- 进程调用
- 块读取-呼叫块写入过程

此外, PMBUS通信还支持组命令。

所有这些命令均在USB接口适配器固件中实现。

B.3 用于PMBUS/SMBUS通信的特殊信号

用于PMBUS/SMBUS通信的特殊信号线参见表2-2。

具体来说, 从设备可以通过将警报信号拉低来向USB接口适配器报告故障或警告; 主机PC则通过轮询USB接口适配器来获取这一警报信号。需要注意的是, USB接口适配器不支持任何主机通知协议, 这意味着它不能作为PMBUS/SMBUS的从设备, 因此无法支持从设备临时转换为主设备并向主机报告故障或警告。

对于PMBUS通信, USB接口适配器提供五个控制信号, 每个信号都可以用来启用/禁用一个或多个从设备的上电。

不支持Write Protect信号, 也不支持RESET信号。

B.4 错误检查码 (PEC) 的实现

默认情况下，USB接口适配器固件对所有不同命令都实现了PEC。但是，PEC字节的实现是可选的。有一个特殊的主机PC到USB接口适配器命令可用于打开或关闭PEC（参见第B.23节）。

如果PEC处于打开或启用状态，USB接口适配器中的固件将向发送到一个或多个从设备的命令附加一个PEC字节，或者检查从从设备接收的PEC字节的有效性。

B.5 通信时钟速度和时钟拉伸

默认时钟速度为100 kHz，与PMBUS 1.0和SMBUS 2.0兼容。为了与未来的PMBUS规范兼容，时钟速度可提升至400kHz（参见第3.9节）。对于每个PMBUS/SMBUS事务，时钟拉伸的累积时间不得超过25 ms。

B.6 ALERT、DATA和CLOCK线的上拉电阻配置

关于此主题的更多详细信息，请参阅第3.8节。

B.7 局限性

由于TUSB3210 USB外围芯片功能有限，因此USB接口适配器存在以下限制：

- 不支持扩展命令。仅支持前256个命令。
- 对于任何块写入或块读取命令，数据字节数限制为仅32字节，尽管PMBUS规范支持每事务最多256字节的数据。
- 不支持地址解析协议（ARP），因此不允许多个主控制器。
- 不支持主机通知协议（请参阅第B.3节）。

B.8 主机PC到USB接口适配器命令列表

表B-1.主机PC到USB接口适配器命令

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
发送字节	0x01	A	B					
接收字节	0x02	A						
写入字节	0x03	A	B	C				
写入Word	0x04	A	B	C	D			
读取字节	0x05	A	B	C				
阅读Word	0x06	A	B	C				
进程调用	0x07	A	B	C	D	E		
块写入	0x08	A	B	C	要写入的C字节数 (从1到32字节)			
区块读取	0x09	A	B	C				
块读取- 块写入过程调用	0x0A	A	B	C	D	要写入的C字节数 (从1到31字节)		
群组命令	0x0B	A	B	C	D (0xFF用于 最后数据包)	要写入的C字节数 (从1到32字节)		如果满足以下 条件 不是最后一个 数据包
断言/取消断言控制线	0x0C	第1行的第0位, ..., 第5行的第4位						位逻辑1表示断言, 位逻辑0表示取消断言
PMBUS电平 信号线	0x0F							
打开/关闭PEC	0x11	0表示关闭, 否则就关闭						

B.9 USB接口适配器到主机PC响应命令列表

表B-2 USB接口适配器到主机PC的响应命令

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
发送字节的响应	0x81	成功: 0 失败: 1						
响应来自 接收字节	0x82	成功: 0 失败: 1	字节读取 来自奴隶					
Write Byte 的响 应	0x83	成功: 0 失败: 1						
Write 的 响 应 Word	0x84	成功: 0 失败: 1						
响应来自 读取字节	0x85	成功: 0 失败: 1	字节读取 来自奴隶					
响应来自 阅读Word	0x86	成功: 0 失败: 1	低字节 已读 来自奴隶	高字节 已读 来自奴隶				
来自进程调用的 响应	0x87	成功: 0 失败: 1	低字节 已读 来自奴隶	高字节 已读 来自奴隶				
块写入响应	0x88	成功: 0 失败: 1						
块读响应	0x89	成功: 0 失败: 1	X	从从机读取的X个字节 (最多32个字节)				
响应来自 块读取- 块写入过程调用	0x8A	成功: 0 失败: 1	X	从从机读取的X个字节 (最多31个字节)				
群组命令响应	0x8B	成功: 0 失败: 1						
响应来自 断言/取消断言控 制线	0x8C	成功: 0 失败: 1						
Poll PMBUS 的 响应 信号线	0x8F	位0用于 控制1, ... 第4位用于 控制5, 第5位用于 警报, 第6和第7位 未使用的						逻辑高电平 1, 逻辑低电平 对应0
对开启/关闭PEC 的响应	0x91	成功: 0 失败: 1						

有关每个命令的详细说明，请参见以下各节。

B.10 发送字节命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
发送字节	0x01	A	B					

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
发送字节的响应	0x81	成功: 0 失败: 1						

以下描述了发送字节命令（带有PEC）的详细位银行：

PMBUS/SMBUS启动→写入A（即左移一位的从设备地址+写入0位）→从设备确认
→写入B（由PMBUS或SMBUS定义的命令字节）→从设备确认→写入PEC字节→从设备确认→PMBUS/
SMBUS停止

B.11 接收字节命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
接收字节	0x02	A						

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
响应来自 接收字节	0x82	成功: 0 失败: 1	字节读取 来自奴隶					

以下描述了带有PEC的接收字节命令的详细位银行：

PMBUS/SMBUS启动→写入A（即左移一位的从设备地址+读取1位+从设备确认→从从设备读取1个数据字
节→主设备确认→从从设备读取PEC字节→主设备NACK→PMBUS/SMBUS停止

B.12 编写字节命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
写入字节	0x03	A	B	C				

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
Write Byte 的响 应	0x83	成功: 0 失败: 1						

以下描述了带有PEC的Write Byte命令的详细位银行：

PMBUS/SMBUS开始→写入A（即左移一位的从设备地址+写入0位）→从设备确认→写入B（由PMBUS或
SMBUS定义的命令字节）→从设备确认→写入C（数据字节）→从设备确认→写入PEC字节→从设备确认
→PMBUS/SMBUS停止

B.13 编写Word命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
写入Word	0x04	A	B	C	D			

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
Write 的 响 应 Word	0x84	成功: 0 失败: 1						

以下描述了带有PEC的Write Word命令的详细位银行:

PMBUS/SMBUS启动→写入A (即左移一位的从设备地址+写入0位) →从设备确认

→写入B (由PMBUS或SMBUS定义的命令字节) →从设备确认→写入C (数据字的低字节) →从设备确认

→写入D (数据字的高字节) →从设备确认→写入PEC字节→从设备确认→PMBUS/SMBUS停止

B.14 读取字节命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
读取字节	0x05	A	B	C				

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
响应来自 读取字节	0x85	成功: 0 失败: 1	字节读取 来自奴隶					

以下描述了带有PEC的Read Byte命令的详细位银行:

PMBUS/SMBUS开始→写入A (即左移一位的从设备地址+写入0位) →从设备确认→写入B (由PMB-

US或SMBUS定义的命令字节) →从设备确认→重复开始→写入C (即左移一位的从设备地址+读取1位)

→从设备确认→从从设备读取1个数据字节→主设备确认→从从设备读取PEC字节→主设备否定确认→

PMBUS/SMBUS停止

B.15 阅读Word命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
阅读Word	0x06	A	B	C				

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
响应来自 阅读Word	0x86	成功: 0 失败: 1	低字节 已读 来自奴隶	高字节 已读 来自奴隶				

以下描述了Read Word命令（带有PEC）的详细位银行：

PMBUS/SMBUS启动→写入A（即左移一位的从设备地址+写入0位）→从设备确认
→写入B（由PMBUS或SMBUS定义的命令字节）→从设备确认→重复启动→写入C（即从设备地址左移一位+读取1的位）→从设备确认→从从设备读取数据字的低字节→主设备确认→从从设备读取数据字的高字节
→主设备确认
→从从机读取PEC字节→主机NACK→PMBUS/SMBUS停止

B.16 进程调用命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
进程调用	0x07	A	B	C	D	E		

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
来自进程调用的 响应	0x87	成功: 0 失败: 1	低字节 已读 来自奴隶	高字节 已读 来自奴隶				

以下描述了带有PEC的Process Call命令的详细位银行：

PMBUS/SMBUS开始→写入A（即左移一位的从设备地址+写入0位）→从设备确认→写入B（由PMBUS或SMBUS定义的命令字节）→从设备确认→写入C（数据字的低字节）→从设备确认→写入D（数据字的高字节）→从设备确认→重复开始→写入E（即左移一位的从设备地址+读取1位）→从设备确认→从从设备读取数据字的低字节→主设备确认→从从设备读取数据字的高字节→主设备确认→从从设备读取PEC字节
→主设备NACK→PMBUS/SMBUS停止

B.17 块写入命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
块写入	0x08	A	B	C	要写入的C字节数（从1到32字节）			

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
块写响应	0x88	成功: 0 失败: 1						

以下描述了带有PEC的Block Write命令的详细位银行:

PMBUS/SMBUS启动→写入A（即左移一位的从设备地址+写入0位）→从设备确认
→写入B（由PMBUS或SMBUS定义的命令字节）→从机确认→写入C（写入从机的字节数）→从机确认→
重复（写下一个字节→从机确认），重复C次
→写入PEC字节→从机确认→PMBUS/SMBUS停止

B.18 块读取命令和响应

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
块读取	0x09	A	B	C				

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
块读响应	0x89	成功: 0 失败: 1	X	从从机读取的X个字节（最多32个字节）				

以下描述了带有PEC的Block Read命令的详细位银行:

PMBUS/SMBUS开始→写入A（即从设备地址左移一位+写入0）→从设备确认→写入B（由PMBUS或SMBUS定义的命令字节）→从设备确认→重复开始→写入C（即从设备地址左移一位+读取1）→从设备确认→读取X（从从设备读取的数据字节数）→主设备确认→重复（读取下一个字节→主设备确认）X次→从设备读取PEC字节→主设备否定确认→PMBUS/SMBUS结束

B.19 块读取-块写入过程调用命令和响应

命令描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
块读取- 块写入过程调用	0x0A	A	B	C	D	要写入的C字节数 (从1到31字节)		

命令描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
响应来自 块读取- 块写入过程调用	0x8A	成功: 0 失败: 1	X	从从机读取X个字节 (最多31个字节)				

以下描述了带有PEC的Block Read-Block Write Process Call命令的详细位银行:

PMBUS/SMBUS启动→写入A (即左移一位的从设备地址+写入0位) →从设备确认
→写入B (由PMBUS或SMBUS定义的命令字节) →从设备确认→写入C (要写入的数据字节数) →从设备确认→重复C次 (每次写入下一个数据字节, 从设备确认) →重复开始→写入D (=从设备地址左移一位+读取1的位) →从设备确认→读取X (要从从设备读取的数据字节数) →主设备确认→重复X次 (每次读取下一个字节, 主设备确认) →从设备读取PEC字节→主设备否定确认→
PMBUS/SMBUS停止

B.20 群组命令和响应

命令描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
群组命令	0x0B	A	B	C	D (0xFF用于 最后数据包)	要写入的C字节数 (从1到32字节)		如果满足以下条件 不是最后一个 数据包

命令描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
群组命令响应	0x8B	成功: 0 失败: 1						

以下描述了带有PEC的组命令的详细位银行:

PMBUS/SMBUS开始→写入A (即左移一位的从设备地址+写入0位) →从设备确认→写入B (由PMBUS或SMBUS定义的命令字节) →从设备确认→写入C (写入从设备的字节数) →从设备确认→重复C次 (每次写下一个字节→从设备确认) →写入PEC字节→从设备确认→如果字节D为0xFF, 则停止PMB/SMBUS; 否则, 等待接收下一个USB数据包以继续向从设备写入数据

B.21 控制线命令与响应的断言/取消断言

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
断言/取消断言控制线	0x0C	第1行的第0位, ..., 第5行的第4位						位逻辑1表示断言, 位逻辑0表示取消断言

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
响应来自断言/取消断言控制线	0x8C	成功: 0 失败: 1						

每个控制线可用于启用/禁用多个从机中的一个的上电。

B.22 PMBUS信号线命令和响应的投票

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 字节3	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
PMBUS信号线的接线	0x0F							

命令 描述	USB数据包 字节0 命令 编码	USB数据包 字节1	USB数据包 字节2	USB数据包 第3字节	USB数据包 字节4	USB数据包 字节5	USB数据包 字节6至63	评论
Poll PMBUS 的响应信号线	0x8F	位0用于控制1, ..., 第4位用于控制5, 第5位用于警报, 第6和第7位未使用的						逻辑高电平1, 逻辑低电平对应0

请注意, 目前不支持RESET信号或WRITE_PROTECT信号。

B.23 打开/关闭PEC命令和响应

命令描述	USB数据包字节0 命令编码	USB数据包字节1	USB数据包字节2	USB数据包字节3	USB数据包字节4	USB数据包字节5	USB数据包字节6至63	评论
打开/关闭PEC	0x11	0为关闭, 否则就关闭						

命令描述	USB数据包字节0 命令编码	USB数据包字节1	USB数据包字节2	USB数据包第3字节	USB数据包字节4	USB数据包字节5	USB数据包字节6至63	评论
对开启/关闭PEC的响应	0x91	成功: 0 失败: 1						

默认设置为启用PEC。为了确保数据完整性，从而提高在开关电源环境中进行PMBUS/SMBUS通信时的电力系统可靠性，TI强烈建议在每次PMBUS/SMBUS通信事务中都采用PEC字节。

重要通知

德州仪器公司及其子公司(TI)保留随时对产品和服务进行更正、修改、增强、改进及其他变更的权利,并有权在不事先通知的情况下停止任何产品或服务。客户在下单前应获取最新相关信息,并确保这些信息的时效性和完整性。所有产品均以TI在订单确认时提供的销售条款和条件为准。

根据TI标准保修条款, TI保证其硬件产品的性能符合销售时适用的规格。在TI认为必要的情况下, 使用测试和其他质量控制技术来支持此保修。除非政府要求强制执行, 否则不一定对每个产品的所有参数进行测试。

德州仪器对应用程序帮助或客户产品设计不承担任何责任。客户应对其使用德州仪器组件的产品和应用程序负责。为了将客户产品和应用程序相关的风险降至最低，客户应提供充分的设计和操作保障。

德州仪器 (TI) 不保证或声明, 任何许可, 无论是明示的还是默示的, 都是在使用 TI 产品或服务的任何组合、机器或工艺下授予的。TI 发布的关于第三方产品或服务的信息, 并不构成 TI 对这些产品或服务的使用许可, 也不构成对其的保证或认可。使用此类信息可能需要从第三方获得专利或其他知识产权的许可, 或从 TI 获得专利或其他知识产权的许可。

TI数据手册或数据表中的信息复制仅允许在不作任何更改的情况下进行，并且必须附带所有相关保证、条件、限制和通知。对这些信息进行更改的复制是一种不公平和欺骗性的商业行为。TI对此类更改的文档不承担任何责任。

如果转售TI产品或服务时声明的内容与TI对该产品或服务所声明的参数不同或超出该参数，则将使所有明示和任何暗示的保证失效，并构成不公平和欺骗性的商业行为。TI对任何此类声明不承担责任。

以下是可获取其他德州仪器产品 and 应用解决方案相关信息的URL:

乘积		应用	
放大器	amplifier.ti.com	音频	www.ti.com/audio
数据转换器	dataconverter.ti.com	汽车	www.ti.com/automotive
数字程控交换机	dsp.ti.com	宽带	www.ti.com/broadband
界面	interface.ti.com	数字控制	www.ti.com/digitalcontrol
逻辑	logic.ti.com	军事的	www.ti.com/military
电源管理	power.ti.com	光网络	www.ti.com/opticalnetwork
微控制器	microcontroller.ti.com	保护措施	www.ti.com/security
低功耗无线	www.ti.com/lpw		www.ti.com/telephony
		电话、视频和成像无线	www.ti.com/video
			www.ti.com/wireless

邮寄地址: Texas Instruments

邮政信箱655303, 达拉斯, 德克萨斯州75265 版权®2006, 德州仪器公司