

电源生产环境中ADP1050和ADP1051 EEPROM的编程与校准

作者: Rico Huang和James Xie

简介

ADP1050和ADP1051内置EEPROM控制器,用于与嵌入式8k×8位EEPROM通信。EEPROM分为两个主要模块:信息模块和主模块。信息模块包含128个8位字节(仅供内部使用),主模块包含8000个8位字节。主模块进一步分为16页。每页含512字节。

主模块的第0页为保留页,用于存储默认命令设置。默认命令设置在ADP1050/ADP1051生产线中预编程。主模块的第1页为保留页,用于存储用户设置。在电源的生产过程中,在该页中编程用户设置。命令编程包括两个主要步骤:将用户设置从十六进制(hex)文件编程到寄存器中,然后使用STORE_USER_ALL命令,把寄存器中的用户设置保存到EEPROM的第1页。

第3页为保留页,用于存储ADP1050/ADP1051 GUI软件的电路板参数设置(电路板设置)。电路板参数设置的编程是可选特性。用户(具体指电源设计师)决定是否编程。编程的方式是用页面写入命令将十六进制文件写入第3页。编程前需要执行页面擦除操作。

本应用笔记详细说明电源生产过程中的ADP1050/ADP1051 EEPROM编程和校准。首先,用于EEPROM编程的十六进制文件是ADP1050/ADP1051 GUI软件在电源开发阶段生成的。其次,利用生成的十六进制文件,可以在电源生产过程中通过EEPROM编程器对ADP1050/ADP1051中的EEPROM进行编程。最后,介绍校准过程。本应用笔记为EEPROM编程和校准提供必要的硬件和软件指引。

全面的详细资料请参阅ADP1050或ADP1051数据手册;数据手册应与本应用笔记配合使用。

目录

简介	1	参考原理图	6
修订历史	2	EEPROM编程	7
生成十六进制文件	3	电序列	7
EEPROM编程方法	5	编程步骤	8
方法1：焊接后编程	5	电源校准和测试	10
方法2：焊接前编程	5	校准和测试的电序列	10
编程方法2的参考原理图	5	校准和测试步骤	11
PMBus规格和时序	5		

修订历史

2014年5月—修订版0：初始版

生成十六进制文件

生产十六进制文件时，请遵循下列步骤：

1. 在电源原型调试完成时，将GUI软件连接到电源板上的 [ADP1050/ADP1051](#)的器件上。
2. 芯片密码解锁之后，在GUI软件中单击生成命令和电路板设置的十六进制文件，如图1所示。通过单击该按钮，所有设置(包括用户设置和电路板设置)都将存储在一个十六进制文件中。在弹出相应的窗口时，指定用于保存该.hex文件的文件夹。
3. 保存生成的十六进制文件。

该十六进制文件，以及BOM、PCB Gerber文件和其他文件将发布给电源生产线。

这个十六进制文件由用户设置和电路板设置构成。用户设置代表PMBus命令的用户设置(从寄存器0x01到寄存器

0xFE6F)。电路板设置代表电路板参数的设置，具体包括以下各项：

- 主变压器的匝数比
- 拓扑结构类型
- 整流器类型
- 输出电感及其DCR(单级或双级)
- 输出电容及其ESR(单级或双级)
- 标称输入电压
- 输入电压检测分压器
- 电流变压器的匝数比
- 输入电流检测电阻
- 标称输出电流
- 输出电压检测分压器
- 输出电流检测寄存器



图1. 生成十六进制文件

12306-001

AN-1306

图2展示了保存着用户命令设置(寄存器0x01至寄存器0xFE6F)和电路板设置的十六进制文件。

示例:

```
:01000100807E          // Register 0x01 value is 1-byte value 0x80.
:0200D300000002B      // Register 0xD3 value is word (2-byte) value 0x0000. In the
                      // ADP1050/ADP1051, this means the EEPROM address offset of 0x0000.
:6300B3000102030060  // Page write to Page 3, totally 99 bytes (0x63).
:00000001FF          // Stop.
```

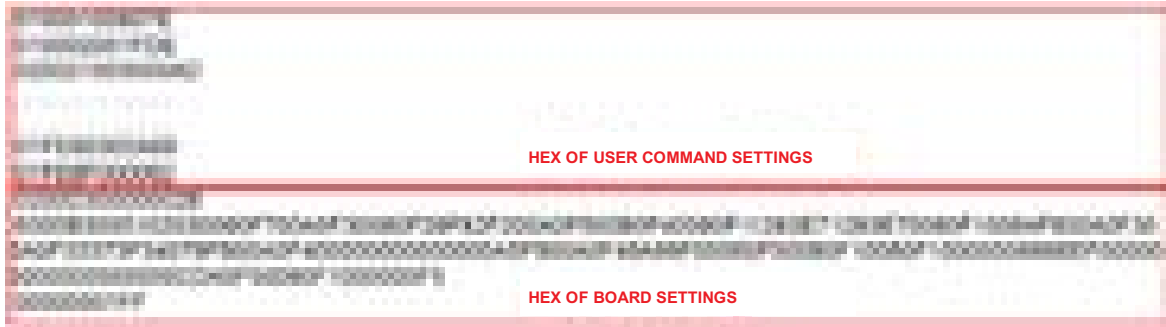


图2. 十六进制文件示例

EEPROM编程方法

主流编程方法有两种：一是在ADP1050/ADP1051器件焊接后通过自动测试设置(ATE)编程；二是在器件焊接前通过EEPROM编程器编程。您也可以使用其他编程方法，比如，系统微控制器，来对EEPROM编程。

方法1：焊接后编程

先构建或订购一套ATE设备。该设备具有EEPROM编程、调整和校准以及系统测试功能。设备包括电脑、直流源、电负载、电压表、电流表、配件和测试夹具。该设备通过GPIO、电源管理总线(PMBus)等通信接口连接。编程软件可以用LabVIEW®、C++或其他类似软件实现。具体步骤如下：

- 焊接PCB板，装配电路板。将ADP1050/ADP1051器件焊接到电路板上。
- 命令ATE设备对器件中的EEPROM编程。
- 命令ATE设备打开直流源和电源单元(PSU)。
- 执行校准和测试。

校准步骤见数据手册的“电源校准和调整”部分。

方法2：焊接前编程

一般而言，应该从编程解决方案提供商那里订购定制的EEPROM编程器，比如Data I/O公司。您也可以自行开发EEPROM编程器。具体步骤如下：

- 从包装中取出ADP1050/ADP1051器件。
- 对器件的EEPROM编程(编程器支持同时对单个器件或多个器件编程)。
- 把器件放回管件中。
- 焊接并装配。

- 使电路板加电，然后执行校准和系统测试。

请注意，如果不需要校准，则这种方法可以大幅提高生产效率。

编程方法2的参考原理图

ADP1050 EEPROM编程的参考原理图如图3所示。ADP1051 EEPROM编程的参考原理图如图4所示。在两幅图中，J1为V_{DD}连接器，J2为PMBus连接器。

说到这些原理图，

1. VDD引脚的去耦电容(C1和C2)以及VCORE引脚的去耦电容(C3)应尽量放在靠近器件之处。
2. 电阻R1(10 kΩ，容差为0.1%)应从ADP1050/ADP1051的RES引脚连接至AGND引脚。
3. 用于SDA和SCL信号的上拉电阻R6和R7 (4.7 kΩ)以及SDA和SCL信号的串联电阻R12和R14 (10 Ω)是可选的。这样可以提高PMBus通信对噪声的免疫能力。

有关该连接的更多信息，请参阅ADP1050或ADP1051数据手册。

电气规格和PMBus通信时序信息请参阅ADP1050和ADP1051数据手册。

PMBUS规格和时序

PMBus协议基于系统管理总线(SMBus)，后者又以2线通信接口内部集成电路(I²C)为基础。PMBus决定了最高总线速率为400 kHz，同时还内置了超时值，这对关键系统来说是至关重要的。与其他标准不同，该标准规定了大量的特定域命令，而不是仅仅规定利用用户定义命令进行通信的方法。

请遵循ADP1050或ADP1051数据手册中规定的参数和PMBus通信时序图。

AN-1306

参考原理图

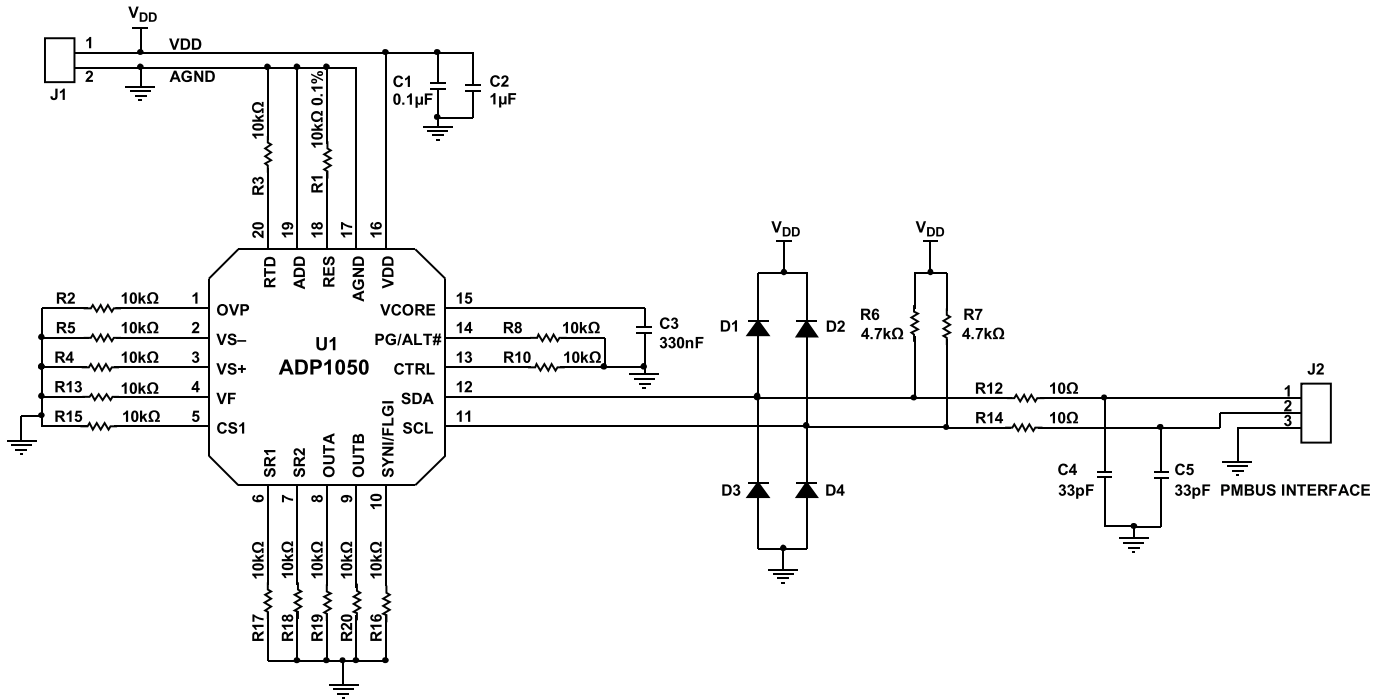


图3. ADP1050 EEPROM编程的参考原理图

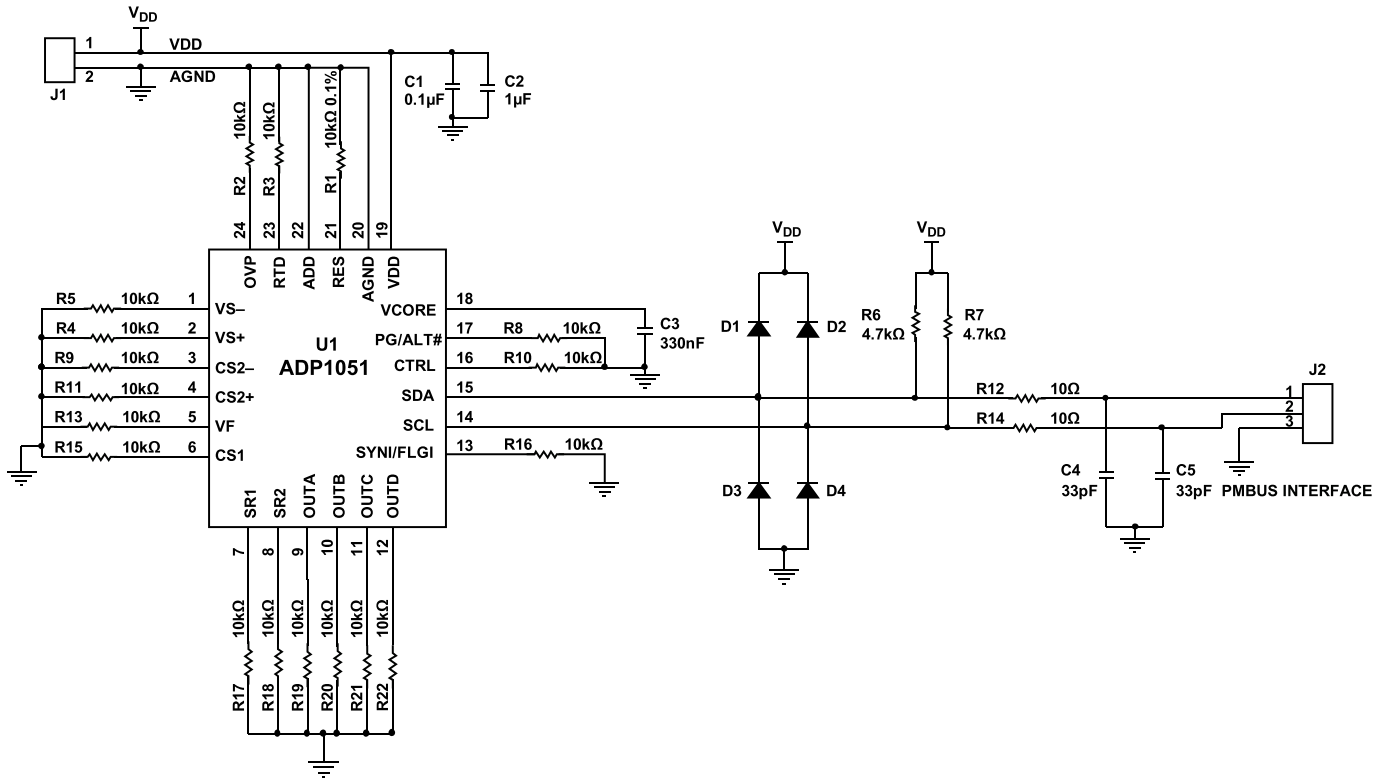


图4. ADP1051 EEPROM编程的参考原理图

EEPROM编程

电序列

图5所示为EEPROM编程的电序列。电路板设置用于对ADP1050或ADP1051 GUI的访问；将电路板设置编程到EEPROM中，此为可选步骤。

用户设置的编程序列

用户设置的编程步骤如下：

1. V_{DD} 上电至3.3 V。在 V_{DD} 达到UVLO时，SCL和SDA信号从低电平拉高，间隔为 t_1 。对间隔 t_1 无特殊要求。在SCL和SDA信号在间隔 t_2 下拉高后，PMBus通信启动。

$$t_1 + t_2 \geq 60 \text{ ms}$$

2. PMBus通信发生在间隔 t_3 中。十六进制文件中用户设置的所有值都被写进ADP1050/ADP1051的寄存器中，EEPROM解锁，可供访问。在 t_3 结束之前，执行STORE_USER_ALL命令(寄存器0x15)，将用户设置保存到EEPROM中。

3. 在间隔 t_4 中，不发生PMBus通信，以便等待EEPROM擦除操作。

$$t_4 \geq 40 \text{ ms}$$

4. EEPROM在间隔 t_5 期间锁定。
5. SCL和SDA信号在经过延迟 t_6 后被拉低。对间隔 t_6 无特殊要求。
6. 在SCL和SDA信号被拉低之后，在延迟 t_7 后， V_{DD} 被关断。对间隔 t_7 无特殊要求。目的是确保 V_{DD} 的关断时间晚于SCL和SDA信号的拉低时间。

7. V_{DD} 电压以及SCL和SDA信号在间隔 t_8 内保持低电平。

$$t_8 = 10 \text{ ms}$$

8. 在 t_8 结束时， V_{DD} 上电。 V_{DD} 在 t_9 期间保持高电平。对间隔 t_9 无特殊要求。在SCL和SDA信号在间隔 t_{10} 下拉高后，PMBus通信启动。

$$t_9 + t_{10} \geq 60 \text{ ms}$$

9. 在间隔 t_{11} 期间，寄存器值被读回，以检查写操作是否正确。
10. 在间隔 t_{12} 期间，SCL和SDA信号保持高电平。对间隔 t_{12} 无特殊要求。

11. SCL和SDA信号被拉低之后，延迟 t_{13} ， V_{DD} 被关断。对间隔 t_{13} 无特殊要求。

电路板设置的编程序列

电路板设置的编程步骤如下：

1. V_{DD} 、SCL和SDA电压电平在间隔 t_{14} 内保持低电平。

$$t_{14} = 10 \text{ ms}$$

2. 然后， V_{DD} 上电至3.3 V。在 V_{DD} 达到UVLO时，SCL和SDA信号从低电平拉高，间隔为 t_{15} 。对间隔 t_{15} 无特殊要求。在SCL和SDA信号在间隔 t_{16} 下拉高后，PMBus通信启动。

$$t_{15} + t_{16} \geq 60 \text{ ms}$$

3. PMBus通信发生在间隔 t_{17} 中。在 t_{17} 结束前，执行EEPROM_PAGE_ERASE命令(寄存器0xD4)，擦除第3页。
4. 在间隔 t_{18} 中，不发生PMBus通信，以便等待EEPROM擦除操作。

$$t_{18} \geq 40 \text{ ms}$$

5. EEPROM写操作发生在 t_{19} 期间。 t_{19} 结束时，EEPROM被锁定。
6. SCL和SDA信号在经过延迟 t_{20} 后被拉低。对间隔 t_{20} 无特殊要求。
7. SCL和SDA信号被拉低之后，延迟 t_{21} ， V_{DD} 被关断。对间隔 t_{21} 无特殊要求。
8. V_{DD} 电压、SCL和SDA信号在间隔 t_{22} 内保持低电平。

$$t_{22} = 10 \text{ ms}$$

9. 在 t_{22} 结束时， V_{DD} 上电。 V_{DD} 在 t_{23} 期间保持高电平。对间隔 t_{23} 无特殊要求。在SCL和SDA信号在间隔 t_{24} 下拉高后，PMBus通信启动。

$$t_{23} + t_{24} \geq 60 \text{ ms}$$

10. 在间隔 t_{25} 期间，电路板设置值被读回，以检查写操作是否正确。
11. 在间隔 t_{26} 期间，SCL和SDA信号保持高电平。对间隔 t_{26} 无特殊要求。
12. SCL和SDA被拉低之后，延迟 t_{27} ， V_{DD} 被关断。对间隔 t_{27} 无特殊要求。

编程步骤

用户设置的编程步骤

1. 将十六进制文件载入编程器的缓冲区。
2. 在间隔 t_3 期间，执行以下操作：
 - a. 把默认芯片密码0xFFFF连续两次写入CHIP_PASSWORD命令(寄存器0xD7)，使芯片解锁。
 - b. 检查寄存器0xFE0[7]是否等于1。如果不等于1，回到步骤a；如果等于1，前往步骤c。
 - c. 此步属于可选步骤。读取ADP1050/ADP1051器件的工厂调整寄存器值，并将其保存在编程器的缓冲区中。目的是避免工厂调整值被覆盖掉。有一个调整密码，可以用来保护工厂调整值，防止其被覆盖掉。调整寄存器为0xFE14、0xFE15、0xFE16、0xFE17、0xFE20、0xFE28、0xFE2A、0xFE2B和0xFE2C。注意，0xFE15、0xFE16和0xFE17仅适用于ADP1051。
 - d. 将十六进制文件中的用户设置值从缓冲区写入相应的寄存器中。

对于ADP1050，将十六进制文件中的用户命令设置值从缓冲区写入相应的寄存器中：

0x01、0x02、0x21、0x24、0x25、0x26、0x27、0x29、0x2A、0x33、0x35、0x36、0x40、0x41、0x44、0x45、0x4F、0x50、0x5E、0x5F、0x60、0x61、0x64、0x99、0x9A、0x9B、0xD8、0xD9、0xFA、0xFB、0xFE00、0xFE01、0xFE02、0xFE03、0xFE05、0xFE06、0xFE07、0xFE08、0xFE09、0xFE0B、0xFE0C、0xFE0D、0xFE0E、0xFE0F、0xFE11、0xFE12、0xFE13、0xFE19、0xFE1A、0xFE1B、0xFE1D、0xFE1E、0xFE1F、0xFE25、0xFE26、0xFE29、0xFE2D、0xFE2F、0xFE30、0xFE31、0xFE32、0xFE33、0xFE38、0xFE39、0xFE3A、0xFE3B、0xFE3C、0xFE3D、0xFE3E、0xFE3F、0xFE40、0xFE41、0xFE42、0xFE43、0xFE4A、0xFE4B、0xFE4C、0xFE4D、0xFE4E、0xFE4F、0xFE50、0xFE52、0xFE53、0xFE54、0xFE55、0xFE57、0xFE58、0xFE59、0xFE61、0xFE62、0xFE63、0xFE64、0xFE65、0xFE67、0xFE68、0xFE69、0xFE6A、0xFE6B、0xFE6C、0xFE6D、0xFE6F

对于ADP1051，将十六进制文件中的用户命令设置值从缓冲区写入相应的寄存器中：

0x01、0x02、0x21、0x24、0x25、0x26、0x27、0x28、0x29、0x2A、0x33、0x35、0x36、0x38、0x40、0x41、0x44、0x45、0x46、0x47、0x48、0x4F、0x50、0x5E、0x5F、0x60、0x61、0x64、0x99、0x9A、0x9B、0xD8、

0xD9、0xFA、0xFB、0xFE00、0xFE01、0xFE02、0xFE03、0xFE05、0xFE06、0xFE07、0xFE08、0xFE09、0xFE0B、0xFE0C、0xFE0D、0xFE0E、0xFE0F、0xFE11、0xFE12、0xFE13、0xFE19、0xFE1A、0xFE1B、0xFE1C、0xFE1D、0xFE1E、0xFE1F、0xFE25、0xFE26、0xFE29、0xFE2D、0xFE2F、0xFE30、0xFE31、0xFE32、0xFE33、0xFE34、0xFE35、0xFE36、0xFE37、0xFE38、0xFE39、0xFE3A、0xFE3B、0xFE3C、0xFE3D、0xFE3E、0xFE3F、0xFE40、0xFE41、0xFE42、0xFE43、0xFE44、0xFE45、0xFE46、0xFE47、0xFE48、0xFE49、0xFE4A、0xFE4B、0xFE4C、0xFE4D、0xFE4E、0xFE4F、0xFE50、0xFE51、0xFE52、0xFE53、0xFE54、0xFE55、0xFE56、0xFE57、0xFE58、0xFE59、0xFE5A、0xFE5B、0xFE5C、0xFE5D、0xFE5E、0xFE5F、0xFE60、0xFE61、0xFE62、0xFE63、0xFE64、0xFE65、0xFE66、0xFE67、0xFE68、0xFE69、0xFE6A、0xFE6B、0xFE6C、0xFE6D、0xFE6E、0xFE6F

- e. 把默认EEPROM密码0xFF连续两次写入EEPROM_PASSWORD命令(寄存器0xD5)，使EEPROM解锁。
 - f. 检查寄存器0xFE2[3]是否等于1。如果不等于1，回到上一步；如果等于1，前往下一步。
 - g. 执行STORE_USER_ALL命令(寄存器0x15)，把用户设置保存到EEPROM中。
3. 等待间隔 t_4 。
 4. 在间隔 t_5 中，把一个不同于0xFF的值写入EEPROM_PASSWORD命令(寄存器0xD5)，以锁定EEPROM。
 5. 关断 V_{DD} ，持续时间为间隔 t_8 。
 6. 再次使 V_{DD} 上电。
 7. 在间隔 t_{11} 期间，执行以下操作：
 - a. 把最新芯片密码(如果未改变，为0xFFFF)连续两次写入CHIP_PASSWORD命令(寄存器0xD7)，使芯片解锁。
 - b. 此步属于可选步骤。如果未执行步骤2c，则跳过此步。读取下列寄存器的调整寄存器值：0xFE14、0xFE15、0xFE16、0xFE17、0xFE20、0xFE28、0xFE2A、0xFE2B和0xFE2C。注意，0xFE15、0xFE16、0xFE17仅适用于ADP1051。将其与第2c步存储在缓冲区中的值进行比较，以确保调整值未被覆盖。
 - c. 读取第2d步中列出的寄存器，比较十六进制文件中的值，确保这些值写入正确。
 8. 在SCL和SDA信号被拉低后，关断ADP1050/ADP1051的 V_{DD} 。

电路板设置的编程步骤

1. 在间隔 t_{17} 中, 执行以下操作:
 - a. 把最新密码(默认为0xFFFF)连续两次写入CHIP_PASSWORD命令(寄存器0xD7), 使芯片解锁。如果芯片密码已更改, 则用最新芯片密码来解锁芯片。
 - b. 检查寄存器0xFE A0[7]是否等于1。如果不等于1, 回到上一步; 如果等于1, 继续下一步。
 - c. 把默认EEPROM密码0xFF连续两次写入EEPROM_PASSWORD命令(寄存器0xD5), 使EEPROM解锁。
 - d. 检查寄存器0xFE A2[3]是否等于1。如果不等于1, 再次回到步骤c; 如果等于1, 前往步骤e。
 - e. 通过EEPROM_PAGE_ERASE命令(寄存器0xD4)擦除第3页。
2. 等待间隔 t_{18} 。
3. 在间隔 t_{19} 中, 执行以下操作:
 - a. 将地址偏移量0x0000写入EEPROM_PAGE_ERASE命令(寄存器0xD3)。
 - b. 用EEPROM_DATA_03命令(寄存器0xB3), 将电路板设置十六进制文件写入EEPROM的第3页, 以连续编程多个字节。
 - c. 写入一个不同于0xFF的值, 以锁定EEPROM。
4. 在间隔 t_{22} 中, 关断 V_{DD} 。
5. 再次使 V_{DD} 上电。
6. 在间隔 t_{25} 中, 执行以下操作:
 - a. 把最新密码(默认为0xFFFF)连续两次写入CHIP_PASSWORD命令(寄存器0xD7), 使芯片解锁。如果芯片密码已更改, 则用最新芯片密码来解锁芯片。
 - b. 从EEPROM的第3页读取电路板设置值, 并将其与电路板设置十六进制文件进行比较, 确保写入操作正确。
 - 将返回数写入EEPROM_NUM_RD_BYTES命令(寄存器0xD2)。
 - 然后, 将地址偏移量写入EEPROM_ADDR_OFFSET命令(寄存器0xD3)。
 - 最后, 用EEPROM_DATA_03命令(寄存器0xB3), 从EEPROM第3页连续读取多个字节。
7. 拉低SCL和SDA信号, 然后, 如果全部值均正确, 则关断 V_{DD} 。

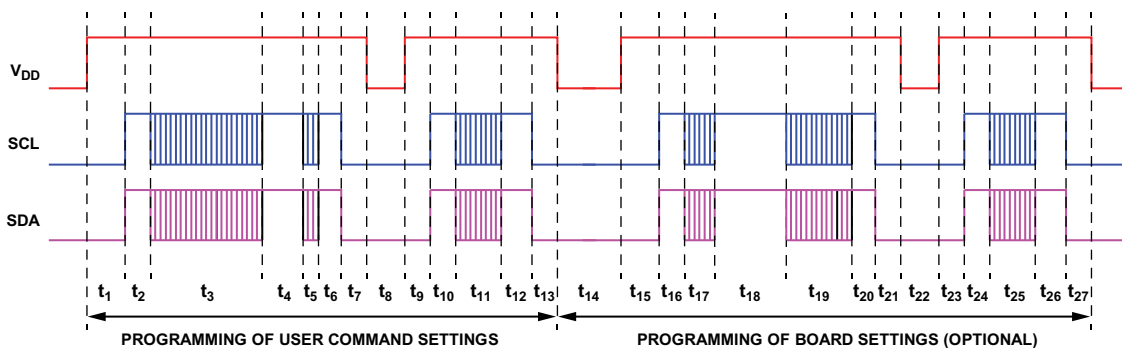


图5. EEPROM编程电序列

电源校准和测试

校准和测试的电序列

所有ADP1050/ADP1051器件均经过工厂调整。如果器件未在电源生产环境中调整，则建议采用容差为0.1%或更好的组件，以达到数据手册中的参数要求。

图6所示为校准电序列。 V_{DD} 源不需要单独施加于PSU之上，因为PSU本身会产生 V_{DD} 。

校准和测试步骤如下：

1. 将输入电压 V_{IN} 施加于PSU上。PSU上电，电压表可以检测到输出电压。在 V_{IN} 达到目标电压后，SCL和SDA信号以 t_1 的间隔从低电平拉高。

$$t_1 \geq 10 \text{ ms}$$

2. 在SCL和SDA信号在间隔 t_2 下拉高后，PMBus通信启动。

$$t_1 + t_2 \geq 70 \text{ ms}$$

3. PMBus通信在间隔 t_3 期间进行。对电流、电压和温度的所有校准均在该步骤处理。EEPROM解锁，可以访问。在 t_3 结束之前，执行STORE_USER_ALL命令(寄存器0x15)，将用户命令设置保存到EEPROM中。
4. 在间隔 t_4 期间，不发生PMBus通信，以便等待EEPROM擦除操作。

$$t_4 \geq 40 \text{ ms}$$

5. EEPROM在间隔 t_5 期间锁定。
6. SCL和SDA信号在经过延迟 t_6 后被拉低。对间隔 t_6 无特殊要求。
7. 在SCL和SDA信号被拉低之后，延迟 t_7 ， V_{IN} 断开。对间隔 t_7 无特殊要求。目的是确保ADP1050/ADP1051的V关断时间晚于SCL和SDA信号的拉低时间。
8. V_{IN} 电压以及SCL和SDA信号在间隔 t_8 内保持低电平。

$$t_8 = 10 \text{ ms}$$

9. 在 t_9 中，将 V_{IN} 施加于PSU。在 V_{IN} 达到目标电压后，SCL和SDA信号以 t_1 的间隔从低电平拉高。

$$t_9 \geq 10 \text{ ms}$$

10. 在SCL和SDA信号在间隔 t_{10} 下拉高后，PMBus通信启动。

$$t_9 + t_{10} \geq 70 \text{ ms}$$

11. 在间隔 t_{11} 期间，对电流、电压和温度的所有测试均在该步骤处理。
12. 在间隔 t_{12} 期间，SCL和SDA信号保持高电平。对间隔 t_{12} 无特殊要求。
13. 在SCL和SDA信号被拉低，延迟 t_{13} ， V_{IN} 与PSU断开。对间隔 t_{13} 无特殊要求。

校准和测试步骤

1. 在间隔 t_3 期间，执行以下操作：
 - a. 把最新密码(默认为0xFFFF)连续两次写入CHIP_PASSWORD命令(寄存器0xD7)，使芯片解锁。
 - b. 检查寄存器0xFE14[7]是否等于1。如果不等于1，回到上一步；如果等于1，继续下一步。
 - c. 把默认调整密码0xFF连续两次写入TRIM_PASSWORD命令(寄存器0xD6)，使调整寄存器解锁，以供写操作访问。
 - d. 读取寄存器0xFE14值N。将值N+1回写到该寄存器。如果该寄存器的返回值不是N+1，则请回到上一步。如果返回值为N+1，将值N写入该寄存器，并继续下一步。
 - e. 执行校准和调整。根据ADP1050/ADP1051数据手册中的“电源校准和调整”部分，执行相应的校准步骤。这些调整寄存器为：0xFE14、0xFE15、0xFE16、0xFE17、0xFE20、0xFE28、0xFE2A、0xFE2B和0xFE2C。注意，0xFE15、0xFE16和0xFE17调整寄存器仅适用于ADP1051。
 - f. 把默认EEPROM密码0xFF连续两次写入EEPROM_PASSWORD命令(寄存器0xD5)，使EEPROM解锁。
 - g. 检查寄存器0xFE2[3]是否等于1。如果不等于1，回到步骤f；如果等于1，继续下一步。
 - h. 执行STORE_USER_ALL命令(寄存器0x15)，把用户命令设置保存到EEPROM中。
2. 等待间隔 t_4 。
3. 在间隔 t_5 中，把一个不同于0xFF的值写入EEPROM_PASSWORD命令(寄存器0xD5)，以锁定EEPROM。
4. 拉低SCL和SDA信号，然后断开 V_{IN} 。
5. 把 V_{IN} 连接到PSU。
6. 在间隔 t_{11} 期间，执行以下操作：
 - a. 把最新芯片密码(默认为0xFFFF)连续两次写入CHIP_PASSWORD命令(寄存器0xD7)，使芯片解锁。
 - b. 通过电压表和电流表收集电压和电流值。
 - c. 通过PMBus命令收集电压和电流读数。
 - d. 如果需要，更改输出电压和输出电流，然后重复前两步中描述的测试。
 - e. 将读数与计量表的值进行比较，以检查读数的准确性。
7. 在SCL和SDA信号被拉低之后，断开 V_{IN} 。

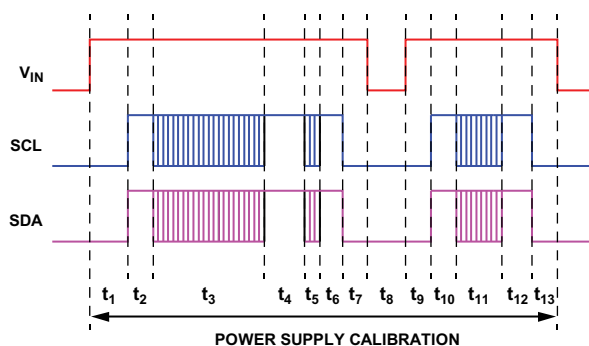


图6. 电源校准和测试的电序列

注释