

摘要

TPS6594-Q1、TPS6593-Q1 和 LP8764-Q1 系列电源管理集成电路 (PMIC) 包含一个可配置的非易失性存储器 (NVM) 空间。借助可扩展 PMIC GUI，可以从通过 GUI 生成或手动生成的 NVM 配置 (汇编文件) 生成二进制映像。本文档详细介绍了硬件设置以及将二进制映像上传到 PMIC NVM 的步骤。

内容

1 引言.....	2
2 硬件和 PMIC 设置.....	3
3 配置概述.....	5
4 指令.....	6
5 特殊注意事项.....	8
5.1 更改串行控制接口.....	8
5.2 更新频率选择.....	8
5.3 PFSM.....	9
5.4 永久锁定 NVM.....	9
5.5 更新寄存器 CRC.....	9
6 NVM 验证.....	12
7 参考文献.....	13
A 由 NVM 支持的寄存器.....	14
B 属于寄存器 CRC 的非 NVM 寄存器.....	16
C 用户寄存器的 CRC，第 0 页和第 4 页.....	18
D I ² C 串行接口示例.....	25
E 修订历史记录.....	26

插图清单

图 1-1. 从可扩展 PMIC GUI 生成二进制文件.....	2
图 3-1. 寄存器映射存储器和 NVM 空间.....	5

表格清单

表 1-1. GUI 二进制输出格式.....	2
表 2-1. TPS6594-Q1 和 TPS6593-Q1 的部分原理图检查清单.....	3
表 2-2. LP8764-Q1 的部分原理图检查清单.....	3
表 4-1. PFSM 地址控制，寄存器地址 0xA4，第 0 页.....	7
表 5-1. 串行接口寄存器.....	8
表 5-2. I ² C 地址和页面关系.....	8
表 5-3. PFSM 子页控制.....	9
表 5-4. 用户寄存器 (第 0 页和第 4 页) 寄存器 CRC 寄存器.....	10
表 6-1. NVM 验证示例.....	12
表 A-1. TPS6594-Q1 和 TPS6593-Q1 NVM 支持的寄存器.....	14
表 A-2. LP876x-Q1 NVM 支持的寄存器.....	14
表 B-1. TPS6594 和 TPS6593 寄存器包含寄存器 CRC 和默认值.....	16
表 B-2. LP876x 寄存器包含寄存器 CRC 和默认值.....	17
表 C-1. 用于用户寄存器 CRC 计算的数据和掩码.....	18
表 D-1. 更新了 I ² C1 地址的 I ² C 示例.....	25

商标

所有商标均为其各自所有者的财产。

1 引言

本文中描述的配置过程写入 NVM 空间，旨在用于生产线或开发环境。该机制不适用于应用，因为该过程会影响稳压器输出和 GPIO 功能。

可扩展 PMIC GUI 提供了一种生成二进制映像的机制，该二进制映像可以上传到 PMIC 的 NVM。选择 **Save as Binary Code**，如图 1-1 所示，从 NVM 编程页面生成二进制文件。有关配置 PMIC 和从该配置生成映像的说明，请参阅 [可扩展 PMIC GUI 用户指南](#)。

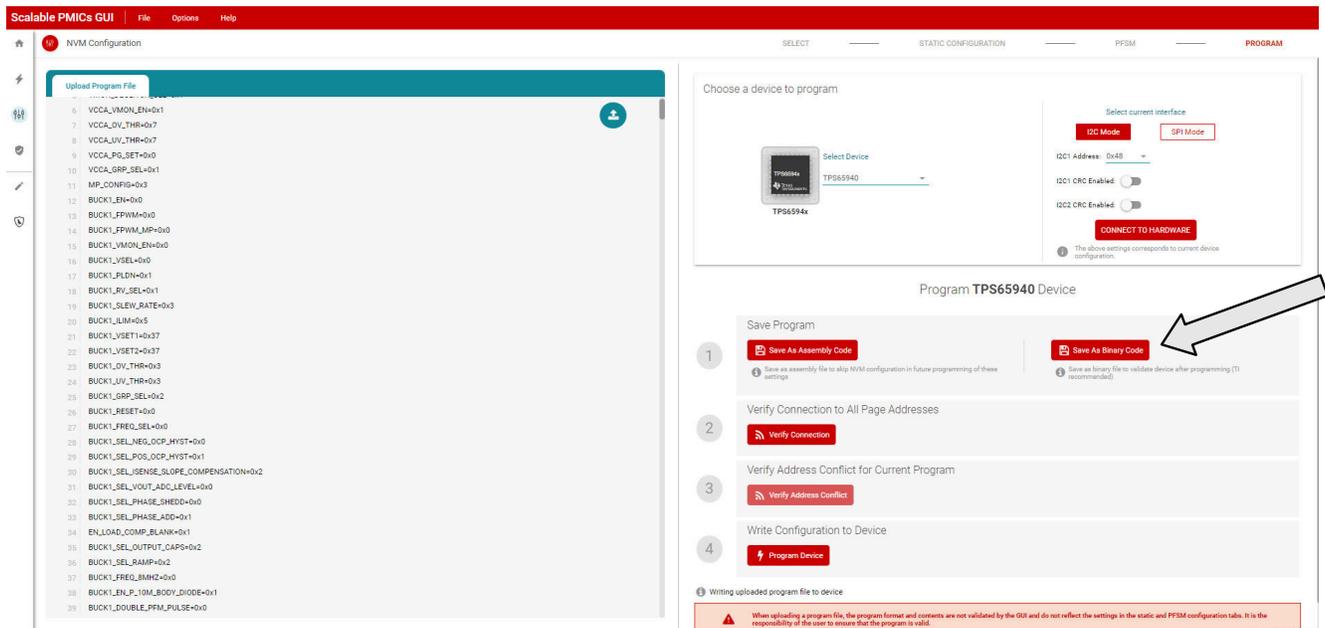


图 1-1. 从可扩展 PMIC GUI 生成二进制文件

表 1-1 显示了二进制文件中的条目示例。每行包含页面信息、寄存器地址和数据。

表 1-1. GUI 二进制输出格式

二进制行	说明
0x0004 = 0xa0	第 0 页条目
0x00d1 = 0x00	
0x0100 = 0x02	第 1 页条目
0x014A = 0x06	
0x3000 = 0x0A	第 3 页条目
0x32ff = 0x00	
0x405 = 0xff	第 4 页条目
0x409 = 0xff	

2 硬件和 PMIC 设置

原理图检查清单中描述了硬件连接 (请参阅参考文献 6) 。表 2-1 和表 2-2 显示了与 NVM 配置相关的所需连接。

表 2-1. TPS6594-Q1 和 TPS6593-Q1 的部分原理图检查清单

引脚	名称	用途	实现正确连接的要求
2	VOUT_LDOVINT	用于噪声去耦电容器的内部 LDO 的输出引脚	电容器 : C _{typ} = 2.2μF ; V _{cap} > 6.3V
3	VOUT_LDOVRTC		
4	VCCA	用于内部 LDO 和其他内部块的模拟输入电压	电容器 : C _{min} = 0.47μF ; C _{typ} = 1μF ; V _{cap} > 6.3V
5	REFGND1	系统基准地	连接到实心接地层, 但不连接顶层的散热焊盘。
6	REFGND2		
30	SDA_I2C1、 SDI_SPI	I ² C 或 SPI 数据	连接到控制器的数据线。对于 I ² C, 请根据速度和 PCB 使用电阻值。
31	SCL_I2C1/SCK_SPI	I ² C 或 SPI 时钟	连接到控制器的时钟线。对于 I ² C, 请根据速度和 PCB 使用电阻值。
32	CS_SPI ¹	SPI 芯片选择	连接到 SPI 控制器的 CS
33	SDO_SPI ¹	SPI SDO	连接到 SPI 控制器的 SDO
48	VIO_IN	数字电源输入, 提供 GPIO 和 I/O 电源电压	电容器 : C _{min} = 0.47μF ; C _{typ} = 1μF ; V _{cap} > 6.3V
57	散热焊盘	电源地, 也是封装的散热焊盘。	连接到顶层电源地多边形网络

(1) 仅 SPI 通信需要引脚 32 和 33。对于 I²C, 仅需要引脚 30 和 31。

备注

有关最新的硬件指南, 请参阅原理图检查清单。

表 2-2. LP8764-Q1 的部分原理图检查清单

引脚	名称	用途	实现正确连接的要求
20	VOUT_LDO	用于噪声去耦电容器的内部 LDO 的输出引脚	电容器 : C _{typ} = 2.2μF ; V _{cap} > 6.3V
18	VCCA	用于内部 LDO 和其他内部块的模拟输入电压	电容器 : C _{min} = 0.1μF ; C _{typ} = 0.47μF ; V _{cap} > 6.3V
19	AGND1	系统基准地	连接到实心接地层, 但不连接顶层的散热焊盘。
21	AGND2		
5	SDA_I2C1、 SDI_SPI	I ² C 或 SPI 数据	连接到控制器的数据线。对于 I ² C, 请根据速度和 PCB 使用电阻值。
4	SCL_I2C1/SCK_SPI	I ² C 或 SPI 时钟	连接到控制器的时钟线。对于 I ² C, 请根据速度和 PCB 使用电阻值。
2	CS_SPI ⁽¹⁾	SPI 芯片选择	连接到 SPI 控制器的 CS
3	SDO_SPI ⁽¹⁾	SPI SDO	连接到 SPI 控制器的 SDO
24	VIO	数字电源输入, 提供 GPIO 和 I/O 电源电压	电容器 : C _{min} = 0.1μF ; C _{typ} = 0.47μF ; V _{cap} > 6.3V
13/29	PGND	电源地	连接到顶层电源地多边形网络

(1) 仅 SPI 通信需要引脚 2 和 3。对于 I²C, 仅需要引脚 4 和 5。

必须应用 VCCA 和 VIO，并且必须可访问串行接口以更新 NVM。VIO 不得连接到或依赖 PMIC 中的任何 GPIO 或稳压器。同样，对于 I²C 串行接口，上拉电压也必须独立于 PMIC。通过 I²C 配置 NVM 时，仅需要 I2C1 接口。I2C2 接口仅在启用时才在应用中使用，并且仅用于看门狗通信。

尝试配置 NVM 之前，还必须了解初始 PMIC 状态。通常，PMIC 必须处于静态或空闲状态。在某些 NVM 配置中，激活 ENABLE 引脚之前，PMIC 不会上电。只将 ENABLE 引脚保持为低电平，即可有效保持 PMIC 处于已知的静态状态。

此外，如果将 NVM 配置为在轮询特定条件的状态之间连续循环，则可能会干扰解锁 NVM 的配置中的初始步骤。

最后，PMIC 允许在安全恢复硬件状态下配置 NVM。这提供了一种可在 NVM 配置错误并导致出错或关断时更改 NVM 的方法。定义 NVM 时，转换到安全恢复状态很重要，不得忽略。可扩展 PMIC GUI 中提供了多种示例模板，包括转换到安全恢复状态的示例。如果未进行转换，则必须处理关联的中断，以便成功解锁 NVM。

3 配置概述

PMIC 有两个存储器空间：寄存器映射空间和 NVM 空间。如图 3-1 所示，配置 NVM 的方法是：首先通过串行接口写入寄存器映射，然后将内容复制到 NVM。由于配置首先需要写入控制稳压器和 GPIO 的寄存器映射，因此必须对 PMIC 资源没有依赖或使用需求。例如，在 PDN-0B 实现⁽⁴⁾中，VIO 由一个受 PMIC GPIO 控制的负载开关供电。由于 GPIO 控制 VIO，对 GPIO 输出的依赖会禁止 NVM 配置，从而影响 I²C 并破坏配置。相比之下，TPS6594 EVM⁽⁵⁾ 为 VIO 提供独立电源。

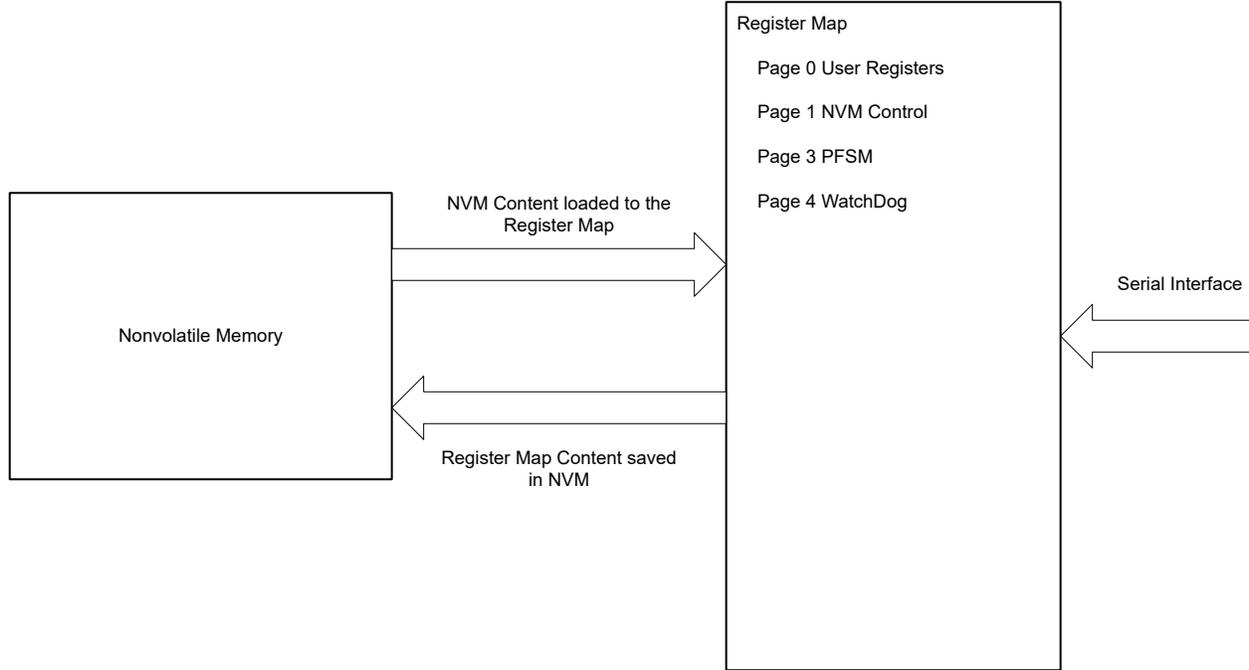


图 3-1. 寄存器映射存储器和 NVM 空间

4 指令

这些指令指的是页面和寄存器地址。当使用 I²C 串行接口时，页面由不同的 I²C 地址描绘，而在 SPI 中，页面信息包含在传输有效载荷的第二个字节中。可将串行接口作为 NVM 配置的一部分进行更改，如节 5 中所述。请参阅有关页面实现的器件数据表。

1. 解锁 NVM

解锁 NVM 需要对位于第 0 页寄存器地址 0xA2 处的 USER_EE_CTRL_1 寄存器进行一系列写入访问。首先将“0x00”写入寄存器以复位解锁机制。接下来的四次写入必须按顺序进行，并且在两次写入之间不能执行寄存器访问。第一次 0x98，第二次 0xB8，第三次 0x13 和第四次 0x7D。成功解锁 NVM 后，将设置第 0 页寄存器地址 0xA3 的位 6 和位 7。

备注

寄存器访问包括 PFSM 和串行接口。在 NVM 解锁期间，PMIC 必须处于空闲状态，并且 PFSM 不会尝试访问寄存器映射。执行解锁的 MCU 还必须将寄存器访问限制为仅与解锁序列关联的写入。在某些 MCU 串行通信中，每次写入后都会自动回读每个寄存器。必须关闭任何此类机制。

2. 禁用 PFSM

解锁 NVM 后，下一步是禁用 PFSM。这是通过设置第 0 页寄存器地址 0xA3 的位 0 来完成的。由于该寄存器还包含解锁内容，因此将 0xC1 写入寄存器。

3. 降压频率和串行接口变更的特殊注意事项

禁用 PFSM 之后，向用户寄存器写入新内容之前，必须应用降压频率或接口变更的任何特殊注意事项。如果降压频率保持不变且串行接口保持不变，则无需额外的指令。请参阅适用的特殊注意事项以及示例。有关详细信息，请参阅节 5.1 和节 5.2

4. 向第 0、1 和 4 页写入内容

写入第 0、1 和 4 页的内容只能包含附录 A 中所述的寄存器地址。如果启用了寄存器 CRC，则非 NVM 寄存器也必须恢复为默认值，如附录 B 中所述。

5. 向第 3 页写入内容

完成第 0、1 和 4 页后，写入第 3 页。第 3 页是一个特殊用例，需要额外处理才能访问 PFSM 存储器空间的子页。页面和子页的描绘也可以在二进制文件中标识，其中第 0 页位于地址空间 0x00-0xFF，第 1 页位于 0x100-0x1FF，第 4 页位于 0x400-0x4FF，第 3 页子页 0 位于 0x3000-0x30FF，子页 1 位于 0x3100-0x32FF，子页 2 位于 0x3200。PFSM 部分对子页进行了详细说明。通过第 0 页寄存器 0xA4 选择第 3 页子页。

CAUTION

TPS6594-Q1 和 TPS6593-Q1 系列器件的第 3 页有三个子页。LP8764-Q1 器件的第 3 页只有两个子页，不存在子页 2。尝试写入 LP8764-Q1 的子页 2 会导致写入子页 1，从而损坏存储器空间。

表 4-1. PFSM 地址控制，寄存器地址 0xA4，第 0 页

位	字段	类型	复位	说明
7-2	保留	R/W	0h	
1-0	PFSM_PAGE_SEL	R/W	0h	选择可寻址的第 3 页地址空间。 0 : 0x000-0x0FF 1 : 0x100-0x1FF 2 : 0x200-0x2FF

6. 锁定 NVM 以防止未来编程 (可选)

如果需要锁定 NVM 以防止未来更新，则第 1 页上的寄存器 0x41 的值必须是 0xA5 以外的值。锁定 NVM 还可防止出于验证目的访问第 3 页。

备注

锁定 NVM 是一项永久性决定，会禁止以后的任何更改。

7. 更新寄存器 CRC

在寄存器映射中更新所有内容后，可以计算和应用寄存器 CRC。只有在所配置的 NVM 中启用寄存器 CRC 时，才需要执行此操作。

8. 将内容从寄存器映射移至 NVM

现在所有用户寄存器内容都已更新且寄存器 CRC 已更新，可将内容复制到 NVM 空间中。要启动内容传输，请设置第 1 页寄存器 0xEF 的位 1。传输不是瞬时的，可以在第 1 页寄存器地址 0xF3 的位 1 中观察到传输的状态。当该位清零时，传输活动就结束了。

5 特殊注意事项

5.1 更改串行控制接口

串行控制接口寄存器为 0x11A、0x122 和 0x123。如表 5-1 所示，0x11A 是串行接口寄存器，用于指示串行接口是 I2C 还是 SPI，以及接口 I2C1 或 SPI 和 I2C2 上是否启用了 CRC。I2C1 地址存储在 0x122 中，I2C2 地址存储在 0x123 中。

表 5-1. 串行接口寄存器

第 1 页，寄存器地址 0x11A 位字段	说明
位 2	<ul style="list-style-type: none"> 0 = I2C2 CRC 禁用 1 = I2C2 CRC 启用
位 1	<ul style="list-style-type: none"> 0 = I2C1 或 SPI CRC 禁用 1 = I2C1 或 SPI CRC 启用
位 0	<ul style="list-style-type: none"> 0 = I2C 模式 1 = SPI 模式

成功解锁 NVM 后，即可更改串行接口。进行更改后，以前的串行接口不再适用，必须适当地更改主机端上的接口。在 I2C 地址更改的情况下，必须对页面进行必要的更新。例如，如果 I2C1 地址从 0x48 更改为 0x28，则更改前后的页面将在表 5-2 中表示。

表 5-2. I2C 地址和页面关系

第 1 页	<ul style="list-style-type: none"> I2C1 地址 = 0x48 I2C2 地址 = 0x12 	<ul style="list-style-type: none"> I2C1 地址 = 0x28 I2C2 地址 = 0x12
第 0 页	0x48	0x28
第 1 页	0x49	0x29
第 3 页	0x4B	0x2B
第 4 页	0x12	0x12

(1) 第 4 页始终与 I2C2 直接关联，第 0 页与 I2C1 直接关联。第 1 页和第 3 页始终从第 0 页偏移。

要从 SPI 更改为 I2C 或从 I2C 更改为 SPI，必须提供必要的 GPIO 来支持这两个接口。提供等待或延迟，以便在继续执行 NVM 指令之前将硬件正确配置为所需的串行接口。有关 I2C 接口和 SPI 的说明，请参阅器件数据表。

备注

在 NVM 解锁期间，不使用 I2C2 物理接口，并且从 PMIC 的 I2C1 (SDA_I2C1、SCL_I2C1) 访问所有页面。

5.2 更新频率选择

降压频率选择受第 1 页寄存器地址 0x18 中的位 3 保护。设置该位之前无法更新降压频率，除非解锁 NVM，否则无法设置该位。完成频率更新后，必须将该位清零。这通常是在写入第 1 页寄存器 0x18 时通过正常程序流程完成。

5.3 PFSM

PFSM 分为多个子页。TPS659x 器件有 3 个子页，而 LP876x 系列器件有 2 个子页。子页的控制 在第 0 页寄存器地址 0xa4 的位 0 和 1 中进行管理。存储在二进制文件中的 PFSM 的内容将移至第 3 页用户寄存器，如表 5-3 所示。

CAUTION

TPS6594-Q1 和 TPS6593-Q1 系列器件的第 3 页有三个子页。LP8764-Q1 器件的第 3 页只有两个子页，不存在子页 2。尝试写入 LP8764-Q1 的子页 2 会导致写入子页 1，从而损坏存储器空间。

表 5-3. PFSM 子页控制

页面, 寄存器地址	第 0 页, 寄存器地址 0xA4		二进制文件
	位 1	位 0	
3, 0x00-0xFF	0	0	0x3000-0x30FF
	0	1	0x3100-0x31FF
	1	0	0x3200-0x32FF

5.4 永久锁定 NVM

要永久锁定 NVM，必须将第 1 页中的寄存器地址 0x41 更改为除 0xA5 以外的某个值。0xA5 是默认值，因此无需任何操作即可将 NVM 保持在解锁状态。当 NVM 永久锁定时，PMIC 功能保持不变，但 NVM 无法再更新。在 NVM 锁定期间，将保留对寄存器映射的访问，如图 3-1 所示。

5.5 更新寄存器 CRC

备注

只有在所编程的 NVM 中启用寄存器 CRC 时，才需要更新寄存器 CRC。已设置第 1 页寄存器 0x18 的位 6。

更新第 3 页和第 1 页寄存器 CRC 内容的过程使用 PMIC 中的 CRC 更新特性。运行 CRC 更新之前，请清除地址 0xF0 到 0xFB 中的寄存器 CRC 内容。设置第 0 页寄存器 0xEF 的位 1 后，将执行 CRC 更新。该位会自行清除。更新完成后，第 0 页地址 0xFB 的内容变为非零值。建议对寄存器 0xFB 执行这项从 0 变为非零值的检查以确认 CRC 完成，然后再尝试继续执行下一步。

PMIC CRC 更新特性计算第 0、1、3 和 4 页的 CRC。第 2 页是受保护的存储空间以及相关的 CRC，本文档未对此进行介绍。在大多数应用中，用户寄存器的计算（第 1 页和第 4 页的组合）不正确。第 0 页寄存器 0x82 的值与默认值不匹配，请参阅附录 B，并且只能从串行接口读取寄存器中的 SPMI_LPM_EN 位和 FORCE_EN_DRV_LOW 位。

备注

如果第 0 页寄存器 0x82 的数据值 ENABLE_DRV_STAT 不是 0x08，则必须手动计算第 0 页和第 4 页的寄存器 CRC，并将结果写入 PMIC。

第 0 页和第 4 页寄存器 CRC 有两个组成部分：包含持续和排除持续。如表 5-4 所示，包含持续 CRC 信息存储在 第 0 页寄存器 0xF0 和 0xF1 中，排除持续存储在 第 0 页寄存器 0xF2 和 0xF3 中。

表 5-4. 用户寄存器 (第 0 页和第 4 页) 寄存器 CRC 寄存器

寄存器	地址, 第 0 页	Bit7-Bit0
CRC_1	0xF0	REGMAP_USER_INCLUDE_PERSIST_CRC16_LOW
CRC_2	0xF1	REGMAP_USER_INCLUDE_PERSIST_CRC16_HIGH
CRC_3	0xF2	REGMAP_USER_EXCLUDE_PERSIST_CRC16_LOW
CRC_4	0xF3	REGMAP_USER_EXCLUDE_PERSIST_CRC16_HIGH

以下代码片段中显示的 CRC 计算取自可扩展 PMIC GUI (请参阅[参考文献 3](#))。附录 C 作为寄存器信息的数组提供。与看门狗相关的信息将映射到适当的位置, 如果不受 NVM 支持, 寄存器将填充默认设置, 如果未定义, 则填充 0x00。

```

const CRC_POLYNOMIAL = 0x755b;
var calculate_register_crc = function(registers, regmap_json, include_persist, page) {
    crc = 0xffff;
    for (var address = 0; address < 0x100; address++) {
        var data;
        var json_address;
        if (address >= 0xf0 && page === 0) {
            // watchdog registers is mapped from 0x0fX in array to 0x40X in register map
            // Offset by 1 since 0x0f0 is mapped to 0x401
            json_address = address + 0x310 + 1;
        } else {
            json_address = address + (page * 0x100);
        }
        if (registers[address] === undefined) {
            if (json_address in regmap_json) {
                // Non-NVM register uses reset value
                data = regmap_json[json_address].reset;
            } else {
                // Register does not exist, so reads 0s
                data = 0;
            }
        } else {
            // Register is NVM-backed, use value from device
            data = registers[address];
        }
        var crc_mask = 0x00;
        if (json_address in regmap_json) {
            if (include_persist) {
                crc_mask = regmap_json[json_address].crc_mask_include_persist;
            } else {
                crc_mask = regmap_json[json_address].crc_mask_exclude_persist;
            }
        }
        data = data & crc_mask;
        crc = crc_d8(crc, data);
    }
    return crc;
};
// Calculate CRC based on look-up table
var crc_d8 = function(crc, data) {
    var table_index = data ^ ((crc & 0xff00) >> 8);
    crc = (crc << 8) ^ lookup_table[table_index];
    return crc & 0xffff;
};
// Compute lookup-table used for CRC calculation
crc = 0x8000;
for (var i = 1; i < 256; i <= 1) {
    if ((crc & 0x8000) !== 0) {
        crc = ((crc << 1) ^ CRC_POLYNOMIAL) & 0xffff;
    } else {
        crc = (crc << 1) & 0xffff;
    }
    for (var j = 0; j < i; j++) {
        lookup_table[i+j] = crc ^ lookup_table[j];
    }
}
    
```

6 NVM 验证

更新 NVM 并进行 PMIC 下电上电后，只需读出用户寄存器映射即可验证 NVM 内容。如果在上电期间 PFSM 指令覆盖寄存器设置，则需要仔细考虑。

或者，可以解锁 NVM，暂停 PFSM，并将 NVM 内容直接传输到用户寄存器。将 NVM 内容传输到用户寄存器后，可通过串行接口再次访问这些值。表 6-1 中描述了这种替代方法。

备注

如果 NVM 锁定，则不可能出现表 6-1 中描述的替代模式。只能验证第 0 页和第 4 页用户寄存器。

表 6-1. NVM 验证示例

指令	I2C 地址 (页)	读/写	寄存器地址	数据	说明
1	0x28 (0)	写入	0xA2	0x00	复位解锁逻辑
2	0x28 (0)	写入	0xA2	0x98	NVM 解锁序列
3	0x28 (0)	写入	0xA2	0xB8	
4	0x28 (0)	写入	0xA2	0x13	
5	0x28 (0)	写入	0xA2	0x7D	
6	0x28 (0)	读取	0xA3	0xC0	
7	0x28 (0)	写入	0xA3	0xC1	暂停 PFSM
8	0x29 (1)	写入	0xEF	0x01	将配置从 NVM 传输到用户寄存器
9	0x29 (1)	读取	0xF3	0x04	确认传输完成；位 1 清零。位 2 无关。
10	0x29 (1)	读取	0x23	0x12	确定要读取的 I2C2 地址
11	0x28 (0)	读取	0x01-0xFF	阵列	从第 0 页读取内容。有效地址范围请参阅附录 A。
12	0x29 (1)	读取	0x00-0xFF	阵列	从第 1 页读取内容。有效地址范围请参阅附录 A。
13	0x12 (4)	读取	0x00-0x0A	阵列	从第 4 页读取内容。有效地址范围请参阅附录 A。
14	0x28 (0)	写入	0xA4	0x00	将 PFSM 设置为子页 0
15	0x2B (3)	读取	0x00-0xFF	阵列	从第 3 页子页 0 读取内容
16	0x28 (0)	写入	0xA4	0x01	将 PFSM 设置为子页 1
17	0x2B (3)	读取	0x00-0xFF	阵列	从第 3 页子页 1 读取内容
18	0x28 (0)	写入	0xA4	0x02	将 PFSM 设置为子页 2
19	0x2B (3)	读取	0x00-0xFF	阵列	从第 3 页子页 2 读取内容
20	0x28 (0)	写入	0xA4	0x00	将 PFSM 设置为子页 0
21	0x28 (0)	写入	0xA2	0x00	复位解锁逻辑

7 参考文献

1. 德州仪器 (TI), [TPS6594-Q1 具有 5 个降压稳压器和 4 个 LDO 且适用于安全相关汽车类应用的电源管理 IC \(PMIC\) 数据表](#)
2. 德州仪器 (TI), [LP8764-Q1 具有集成开关的四相、20A 降压转换器 数据表](#)
3. 德州仪器 (TI), [可扩展 PMIC GUI 用户指南](#)
4. 德州仪器 (TI), [适用于 Jacinto 7 DRA829 和 TDA4VM 汽车 PDN-0B 的双 TPS6594-Q1 PMIC 用户指南](#)
5. 德州仪器 (TI), [TPS6594x-Q1 评估模块 用户指南](#)
6. 德州仪器 (TI), [TPS6594-Q1 原理图 PCB 检查清单 \(Rev A\) 应用手册](#)

A 由 NVM 支持的寄存器

虽然 NVM 空间是连续的，但只对表 A-1 中所述的寄存器进行写入以更新 NVM。

表 A-1. TPS6594-Q1 和 TPS6593-Q1 NVM 支持的寄存器

第 0 页	第 1 ⁴ 页	第 3 ² 页	第 4 页
0x04-0x20	0x01	0x00-0xFF	0x05
0x23-0x2C	0x03-0x0A	0x00-0xFF	0x09
0x31-0x3E	0x0C-0x14	0x00-0xFF	
0x41-0x54	0x16-0x28 ¹		
0x56-0x59	0x33-0x35		
0x78-0x7E	0x3F-0x40 ⁵		
0x84	0x42-0x43		
0x87-0x88	0x45-0x4A		
0x8A-0x8E			
0x92			
0x9B			
0xA7			
0xC3			
0xCD-0xD1			
0xF0-0xFB ³			

- (1) 当更改串行接口时，建议在解锁 NVM 后立即处理第 1 页地址 0x18、0x22 和 0x23。
- (2) PFSM 中描述了第 3 页子页地址控制。
- (3) 存储在地址 0xF0 到 0xFB 中的内容与寄存器 CRC 相关，更新寄存器 CRC 中对此进行了描述。
- (4) 第 1 页寄存器 0x00、0x02、0x0B、0x15、0xA8、0xB1、0xBA、0xC3 和 0xCC 的出厂设置不可更改。
- (5) 节 5.4 中描述了第 1 页寄存器 0x41

表 A-2. LP876x-Q1 NVM 支持的寄存器

第 0 页	第 1 ⁴ 页	第 3 ² 页	第 4 页
0x04-0x0B	0x01	0x00-0xFF	0x05
0x0E-0x15	0x03-0x0A	0x00-0xFF	0x09
0x18-0x1B	0x0C-0x14		
0x2B-0x3A	0x16-0x28 ¹		
0x3C-0x3E	0x33-0x37		
0x41	0x3F-0x40 ⁵		
0x43-0x4A	0x42-0x43		
0x4E-0x54			
0x56-0x59			
0x78			
0x7B-0x7D			
0x84			
0x87-0x88			
0x8A-0x8B			
0x8E			
0x92			

表 A-2. LP876x-Q1 NVM 支持的寄存器 (continued)

第 0 页	第 1 ⁴ 页	第 3 ² 页	第 4 页
0xA7-0xA8			
0xC3			
0xCD-0xD0			
0xF0-0xF9 ³			

- (1) 当更改串行接口时，建议在解锁 NVM 后立即处理第 1 页地址 0x18、0x22 和 0x23。
- (2) [PFSM](#) 中描述了第 3 页子页地址控制。
- (3) 存储在地址 0xF0 到 0xF9 中的内容与寄存器 CRC 相关，[更新寄存器 CRC](#) 中对此进行了描述。
- (4) 第 1 页寄存器 0x00、0x02、0x0B、0x15、0xA8、0xB1、0xBA 和 0xC3 的出厂设置不可更改。
- (5) [节 5.4](#) 中描述了第 1 页寄存器 0x41

B 属于寄存器 CRC 的非 NVM 寄存器

更新寄存器 CRC 前，表 B-1 和表 B-2 中的寄存器必须恢复为默认值。此过程仅在启用寄存器 CRC 时才适用。

表 B-1. TPS6594 和 TPS6593 寄存器包含寄存器 CRC 和默认值

第 0 页		第 4 页	
地址	默认	地址	默认
0x22	0x00	0x01	0x00
0x80	0x00	0x03	0x7F
0x81	0x00	0x04	0x7F
0x82	0x08 ¹	0x07	0x0A
0x85	0x00		
0x86	0x00		
0x8F	0x00		
0x90	0x00		
0x91	0x00		
0x92	0x00		
0x93	0x00		
0x94	0x00		
0x96	0x00		
0x98	0x00		
0x99	0x00		
0x9A	0x00		
0x9B	0x00		
0x9C	0x00		
0x9D	0x00		
0x9E	0x00		
0x9F	0x00		
0xAB	0x00		
0xC9	0x00		
0xCA	0x00		
0xCB	0x00		
0xCC	0x00		

- (1) 该寄存器在串行接口中为只读模式。如果寄存器的值与默认值不匹配，则必须从外部计算第 0 页和第 4 页的寄存器 CRC，并将结果编程到 PMIC 中。

表 B-2. LP876x 寄存器包含寄存器 CRC 和默认值

第 0 页		第 4 页	
地址	默认	寄存器	默认
0x80	0x00	0x01	0x00
0x81	0x00	0x03	0x7F
0x82	0x08 ¹	0x04	0x7F
0x85	0x00	0x07	0x0A
0x86	0x00		
0x8F	0x00		
0x90	0x00		
0x91	0x00		
0x92	0x00		
0x93	0x00		
0x94	0x00		
0x96	0x00		
0xAB	0x00		
0xC9	0x00		
0xCA	0x00		
0xCB	0x00		
0xCC	0x00		

- (1) 该寄存器在串行接口中为只读模式。如果寄存器的值与默认值不匹配，则必须从外部计算第 0 页和第 4 页的寄存器 CRC，并将结果编程到 PMIC 中。

C 用户寄存器的 CRC，第 0 页和第 4 页

表 C-1 包括数据或数据引用以及节 5.5 中所述的适当掩码。

表 C-1. 用于用户寄存器 CRC 计算的数据和掩码

地址	TP659x-Q1 数据	LP876x-Q1 数据	TPS659x-Q1 包含掩码	TPS659x-Q1 排除掩码	LP876x-Q1 包含掩码	LP876x-Q1 排除掩码
0x00	0x00	0x00	NA ³	NA ³	NA ³	NA ³
0x01 ¹	从器件读取	从器件读取	0xFF	0xFF	0xFF	0xFF
0x02 ¹	从器件读取	从器件读取	0xFF	0xFF	0xFF	0xFF
0x03 ¹	从器件读取	从器件读取	0xFF	0xFF	0xFF	0xFF
0x04	NVM ²	NVM	0xFF	0xBF	0xFF	0xBF
0x05	NVM	NVM	0xFF	0x3F	0xFF	0x3F
0x06	NVM	NVM	0xFF	0xBB	0xFF	0xBB
0x07	NVM	NVM	0xFF	0x3F	0xFF	0x3F
0x08	NVM	NVM	0xFF	0xBF	0xFF	0xBF
0x09	NVM	NVM	0xFF	0x3F	0xFF	0x3F
0x0A	NVM	NVM	0xFF	0xBB	0xFF	0xBB
0x0B	NVM	NVM	0xFF	0x3F	0xFF	0x3F
0x0C	NVM	0x00	0xFF	0xBB	不适用	不适用
0x0D	NVM	0x00	0xFF	0x3F	不适用	不适用
0x0E	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x0F	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x10	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x11	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x12	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x13	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x14	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x15	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x16	NVM	0x00	0xFF	0xFF	不适用	不适用
0x17	NVM	0x00	0xFF	0xFF	不适用	不适用
0x18	NVM	NVM	0xFF	0x3F	0xFF	0x3F
0x19	NVM	NVM	0xFF	0x3F	0xFF	0x3F
0x1A	NVM	NVM	0xFF	0x3F	0xFF	0x3F
0x1B	NVM	NVM	0xFF	0x3F	0xFF	0x3F
0x1C	NVM	0x00	0xFF	0x3F	不适用	不适用
0x1D	NVM	0x00	0xFF	0xF3	不适用	不适用
0x1E	NVM	0x00	0xFF	0xF3	不适用	不适用
0x1F	NVM	0x00	0xFF	0xF3	不适用	不适用
0x20	NVM	0x00	0xFF	0xF3	不适用	不适用
0x21	0x00	0x00	0x00	0x00	0x00	0x00
0x22	0x00	0x00	0xFF	0x01	不适用	不适用
0x23	NVM	0x00	0xFF	0xFE	不适用	不适用
0x24	NVM	0x00	0xFF	0xFE	不适用	不适用

表 C-1. 用于用户寄存器 CRC 计算的数据和掩码 (continued)

地址	TP659x-Q1 数据	LP876x-Q1 数据	TPS659x-Q1 包含掩码	TPS659x-Q1 排除掩码	LP876x-Q1 包含掩码	LP876x-Q1 排除掩码
0x25	NVM	0x00	0xFF	0xFE	不适用	不适用
0x26	NVM	0x00	0xFF	0x7F	不适用	不适用
0x27	NVM	0x00	0xFF	0x3F	不适用	不适用
0x28	NVM	0x00	0xFF	0x3F	不适用	不适用
0x29	NVM	0x00	0xFF	0x3F	不适用	不适用
0x2A	NVM	0x00	0xFF	0x3F	不适用	不适用
0x2B	NVM	NVM	0xFF	0x21	0xFF	0xFF
0x2C	NVM	NVM	0xFF	0x7F	0xFF	0x7F
0x2D	0x00	NVM	不适用 ⁽³⁾	不适用 ⁽³⁾	0xFF	0x7F
0x2E	0x00	NVM	不适用	不适用	0xFF	0xFF
0x2F	0x00	NVM	不适用	不适用	0xFF	0x7F
0x30	0x00	NVM	不适用	不适用	0xFF	0xFF
0x31	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x32	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x33	NVM	NVM	0xFF	0x10	0xFF	0xFF
0x34	NVM	NVM	0xFF	0x10	0xFF	0xFF
0x35	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x36	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x37	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x38	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x39	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x3A	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x3B	NVM	0x00	0xFF	0xFF	不适用	不适用
0x3C	NVM	NVM	0xFF	0x11	0xFF	0x00
0x3D	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x3E	NVM	NVM	0xFF	0x07	0xFF	0x03
0x3F	0x00	0x00	不适用	不适用	不适用	不适用
0x40	0x00	0x00	不适用	不适用	不适用	不适用
0x41	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x42	NVM	0x00	0xFF	0xFF	不适用	不适用
0x43	NVM	NVM	0xFF	0x0F	0xFF	0xFC
0x44	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x45	NVM	NVM	0xFF	0x03	0xFF	0x03
0x46	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x47	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x48	NVM	NVM	0xFF	0x3F	0xFF	0x0F
0x49	NVM	NVM	0xFF	0xBB	0xFF	0xBB
0x4A	NVM	NVM	0xFF	0xBB	0xFF	0xBB
0x4B	NVM	0x00	0xFF	0x0B	不适用	不适用

表 C-1. 用于用户寄存器 CRC 计算的数据和掩码 (continued)

地址	TP659x-Q1 数据	LP876x-Q1 数据	TPS659x-Q1 包含掩码	TPS659x-Q1 排除掩码	LP876x-Q1 包含掩码	LP876x-Q1 排除掩码
0x4C	NVM	0x00	0xFF	0xBB	不适用	不适用
0x4D	NVM	0x00	0xFF	0xBB	不适用	不适用
0x4E	NVM	NVM	0xFF	0x03	0xFF	0x6F
0x4F	NVM	NVM	0xFF	0xF3	0xFF	0xFF
0x50	NVM	NVM	0xFF	0xF3	0xFF	0xFF
0x51	NVM	NVM	0xFF	0x3F	0xFF	0x1B
0x52	NVM	NVM	0xFF	0x20	0xFF	0x00
0x53	NVM	NVM	0xFF	0x0B	0xFF	0x0B
0x54	NVM	NVM	0xFF	0xD6	0xFF	0xD6
0x55	0x00	0x00	不适用	不适用	不适用	不适用
0x56	NVM	NVM	0xFF	0x0F	0xFF	0x8F
0x57	NVM	NVM	0xFF	0xAB	0xFF	0xAB
0x58	NVM	NVM	0xFF	0x09	0xFF	0x09
0x59	NVM	NVM	0xFF	0x3F	0xFF	0x38
0x5A	0x00	0x00	不适用	不适用	不适用	不适用
0x5B	0x00	0x00	不适用	不适用	不适用	不适用
0x5C	0x00	0x00	不适用	不适用	不适用	不适用
0x5D	0x00	0x00	不适用	不适用	不适用	不适用
0x5E	0x00	0x00	不适用	不适用	不适用	不适用
0x5F	0x00	0x00	不适用	不适用	不适用	不适用
0x60	0x00	0x00	不适用	不适用	不适用	不适用
0x61	0x00	0x00	不适用	不适用	不适用	不适用
0x62	0x00	0x00	不适用	不适用	不适用	不适用
0x63	0x00	0x00	不适用	不适用	不适用	不适用
0x64	0x00	0x00	不适用	不适用	不适用	不适用
0x65	0x00	0x00	不适用	不适用	不适用	不适用
0x66	0x00	0x00	不适用	不适用	不适用	不适用
0x67	0x00	0x00	不适用	不适用	不适用	不适用
0x68	0x00	0x00	不适用	不适用	不适用	不适用
0x69	0x00	0x00	不适用	不适用	不适用	不适用
0x6A	0x00	0x00	不适用	不适用	不适用	不适用
0x6B	0x00	0x00	不适用	不适用	不适用	不适用
0x6C	0x00	0x00	不适用	不适用	不适用	不适用
0x6D	0x00	0x00	不适用	不适用	不适用	不适用
0x6E	0x00	0x00	不适用	不适用	不适用	不适用
0x6F	0x00	0x00	不适用	不适用	不适用	不适用
0x70	0x00	0x00	不适用	不适用	不适用	不适用
0x71	0x00	0x00	不适用	不适用	不适用	不适用
0x72	0x00	0x00	不适用	不适用	不适用	不适用

表 C-1. 用于用户寄存器 CRC 计算的数据和掩码 (continued)

地址	TP659x-Q1 数据	LP876x-Q1 数据	TPS659x-Q1 包含掩码	TPS659x-Q1 排除掩码	LP876x-Q1 包含掩码	LP876x-Q1 排除掩码
0x73	0x00	0x00	不适用	不适用	不适用	不适用
0x74	0x00	0x00	不适用	不适用	不适用	不适用
0x75	0x00	0x00	不适用	不适用	不适用	不适用
0x76	0x00	0x00	不适用	不适用	不适用	不适用
0x77	0x00	0x00	不适用	不适用	不适用	不适用
0x78	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0x79	NVM	0x00	0xFF	0x03	不适用	不适用
0x7A	NVM	0x00	0xFF	0xFF	不适用	不适用
0x7B	NVM	NVM	0xFF	0xF9	0xFF	0xFF
0x7C	NVM	NVM	0xFF	0x03	0xFF	0x03
0x7D	NVM	NVM	0xFF	0xFB	0xFF	0xFB
0x7E	NVM	0x00	0x7F	0x0F	不适用	不适用
0x7F	0x00	0x00	不适用	不适用	不适用	不适用
0x80	0x00	0x00	0xFF	0x01	0xFF	0x01
0x81	0x00	0x00	0xFF	0xFF	0xFF	0xFF
0x82	0x08	0x08	0xF8	0x18	0xF8	0x18
0x83	0x00	0x00	不适用	不适用	不适用	不适用
0x84	NVM	NVM	0xEF	0x0F	0xEF	0x0F
0x85	0x00	0x00	0xF0	0xF0	0xF0	0xF0
0x86	0x00	0x00	0xFF	0x03	0xFF	0x03
0x87	NVM	NVM	0xFF	0x1F	0xFF	0x0F
0x88	NVM	NVM	0xFF	0x07	0xFF	0x07
0x89	0x00	0x00	不适用	不适用	不适用	不适用
0x8A	NVM	NVM	0xFF	0x1F	0xFF	0xFF
0x8B	NVM	NVM	0xFF	0x1F	0xFF	0x1F
0x8C	NVM	0x00	0xFF	0xFF	不适用	不适用
0x8D	NVM	0x00	0xFF	0xFF	不适用	不适用
0x8E	NVM	NVM	0xFF	0x0F	0xFF	0x0F
0x8F	0x00	0x00	0xFF	0x01	0xFF	0x01
0x90	0x00	0x00	0xFF	0xFF	0xFF	0xFF
0x91	0x00	0x00	0xFF	0xFF	0xFF	0xFF
0x92 ⁽⁴⁾	0x00	0x00	0xFF	0xEF	0xFF	0xEF
0x93	0x00	0x00	0xFF	0xFF	0xFF	0xFF
0x94	0x00	0x00	0xFF	0xFF	0xFF	0xFF
0x95	0x00	0x00	0xFF	0xFF	0xFF	0xFF
0x96	0x00	0x00	0xFF	0xFF	0xFF	0xFF
0x97	0x00	0x00	不适用	不适用	不适用	不适用
0x98	0x00	0x00	0xFF	0x01	不适用	不适用
0x99	0x00	0x00	0xFF	0xFF	不适用	不适用

表 C-1. 用于用户寄存器 CRC 计算的数据和掩码 (continued)

地址	TP659x-Q1 数据	LP876x-Q1 数据	TPS659x-Q1 包含掩码	TPS659x-Q1 排除掩码	LP876x-Q1 包含掩码	LP876x-Q1 排除掩码
0x9A	0x00	0x00	0xFF	0xFF	不适用	不适用
0x9B ⁽⁴⁾	0x00	0x00	0xFF	0xEF	不适用	不适用
0x9C	0x00	0x00	0xFF	0xFF	不适用	不适用
0x9D	0x00	0x00	0xFF	0xFF	不适用	不适用
0x9E	0x00	0x00	0xFF	0xFF	不适用	不适用
0x9F	0x00	0x00	0xFF	0xFF	不适用	不适用
0xA0	0x00	0x00	不适用	不适用	不适用	不适用
0xA1	0x00	0x00	不适用	不适用	不适用	不适用
0xA2	0x00	0x00	不适用	不适用	不适用	不适用
0xA3	0x00	0x00	不适用	不适用	不适用	不适用
0xA4	0x00	0x00	不适用	不适用	不适用	不适用
0xA5	0x00	0x00	不适用	不适用	不适用	不适用
0xA6	0x00	0x00	不适用	不适用	不适用	不适用
0xA7	NVM	NVM	0xFF	0xFF	0xFF	0xFF
0xA8	0x00	NVM	不适用	不适用	0xFF	0x3F
0xA9	0x00	0x00	不适用	不适用	不适用	不适用
0xAA	0x00	0x00	不适用	不适用	不适用	不适用
0xAB	0x00	0x00	不适用	不适用	不适用	不适用
0xAC	0x00	0x00	不适用	不适用	不适用	不适用
0xAD	0x00	0x00	不适用	不适用	不适用	不适用
0xAE	0x00	0x00	不适用	不适用	不适用	不适用
0xAF	0x00	0x00	不适用	不适用	不适用	不适用
0xB0	0x00	0x00	不适用	不适用	不适用	不适用
0xB1	0x00	0x00	不适用	不适用	不适用	不适用
0xB2	0x00	0x00	不适用	不适用	不适用	不适用
0xB3	0x00	0x00	不适用	不适用	不适用	不适用
0xB4	0x00	0x00	不适用	不适用	不适用	不适用
0xB5	0x00	0x00	不适用	不适用	不适用	不适用
0xB6	0x00	0x00	不适用	不适用	不适用	不适用
0xB7	0x00	0x00	不适用	不适用	不适用	不适用
0xB8	0x00	0x00	不适用	不适用	不适用	不适用
0xB9	0x00	0x00	不适用	不适用	不适用	不适用
0xBA	0x00	0x00	不适用	不适用	不适用	不适用
0xBB	0x00	0x00	不适用	不适用	不适用	不适用
0xBC	0x00	0x00	不适用	不适用	不适用	不适用
0xBD	0x00	0x00	不适用	不适用	不适用	不适用
0xBE	0x00	0x00	不适用	不适用	不适用	不适用
0xBF	0x00	0x00	不适用	不适用	不适用	不适用
0xC0	0x00	0x00	不适用	不适用	不适用	不适用

表 C-1. 用于用户寄存器 CRC 计算的数据和掩码 (continued)

地址	TP659x-Q1 数据	LP876x-Q1 数据	TPS659x-Q1 包含掩码	TPS659x-Q1 排除掩码	LP876x-Q1 包含掩码	LP876x-Q1 排除掩码
0xC1	0x00	0x00	不适用	不适用	不适用	不适用
0xC2	0x00	0x00	不适用	不适用	不适用	不适用
0xC3 ⁽⁴⁾	NVM	NVM	0xFF	0x00	0xFF	0x00
0xC4	0x00	0x00	不适用	不适用	不适用	不适用
0xC5	0x00	0x00	不适用	不适用	不适用	不适用
0xC6	0x00	0x00	不适用	不适用	不适用	不适用
0xC7	0x00	0x00	不适用	不适用	不适用	不适用
0xC8	0x00	0x00	不适用	不适用	不适用	不适用
0xC9	0x00	0x00	0xFF	0x00	0xFF	0x00
0xCA	0x00	0x00	0xFF	0x00	0xFF	0x00
0xCB	0x00	0x00	0xFF	0x00	0xFF	0x00
0xCC	0x00	0x00	0xFF	0x00	0xFF	0x00
0xCD	NVM	NVM	0xFF	0x00	0xFF	0x00
0xCE	NVM	NVM	0xFF	0x00	0xFF	0x00
0xCF	NVM	NVM	0xFF	0x00	0xFF	0x00
0xD0	NVM	NVM	0xFF	0x00	0xFF	0x00
0xD1	NVM	0x00	0xFF	0x00	不适用	不适用
0xD2	0x00	0x00	不适用	不适用	不适用	不适用
0xD3	0x00	0x00	不适用	不适用	不适用	不适用
0xD4	0x00	0x00	不适用	不适用	不适用	不适用
0xD5	0x00	0x00	不适用	不适用	不适用	不适用
0xD6	0x00	0x00	不适用	不适用	不适用	不适用
0xD7	0x00	0x00	不适用	不适用	不适用	不适用
0xD8	0x00	0x00	不适用	不适用	不适用	不适用
0xD9	0x00	0x00	不适用	不适用	不适用	不适用
0xDA	0x00	0x00	不适用	不适用	不适用	不适用
0xDB	0x00	0x00	不适用	不适用	不适用	不适用
0xDC	0x00	0x00	不适用	不适用	不适用	不适用
0xDD	0x00	0x00	不适用	不适用	不适用	不适用
0xDE	0x00	0x00	不适用	不适用	不适用	不适用
0xDF	0x00	0x00	不适用	不适用	不适用	不适用
0xE0	0x00	0x00	不适用	不适用	不适用	不适用
0xE1	0x00	0x00	不适用	不适用	不适用	不适用
0xE2	0x00	0x00	不适用	不适用	不适用	不适用
0xE3	0x00	0x00	不适用	不适用	不适用	不适用
0xE4	0x00	0x00	不适用	不适用	不适用	不适用
0xE5	0x00	0x00	不适用	不适用	不适用	不适用
0xE6	0x00	0x00	不适用	不适用	不适用	不适用
0xE7	0x00	0x00	不适用	不适用	不适用	不适用

表 C-1. 用于用户寄存器 CRC 计算的数据和掩码 (continued)

地址	TP659x-Q1 数据	LP876x-Q1 数据	TPS659x-Q1 包含掩码	TPS659x-Q1 排除掩码	LP876x-Q1 包含掩码	LP876x-Q1 排除掩码
0xE8	0x00	0x00	不适用	不适用	不适用	不适用
0xE9	0x00	0x00	不适用	不适用	不适用	不适用
0xEA	0x00	0x00	不适用	不适用	不适用	不适用
0xEB	0x00	0x00	不适用	不适用	不适用	不适用
0xEC	0x00	0x00	不适用	不适用	不适用	不适用
0xED	0x00	0x00	不适用	不适用	不适用	不适用
0xEE	0x00	0x00	不适用	不适用	不适用	不适用
0xEF	0x00	0x00	不适用	不适用	不适用	不适用
0xF0	0x00	0x00	0xFF	0xFF	0xFF	0xFF
0xF1	0x00	0x00	不适用	不适用	不适用	不适用
0xF2	0x7F	0x7F	0xFF	0x7F	0xFF	0x7F
0xF3	0x7F	0x7F	0xFF	0x7F	0xFF	0x7F
0xF4	第 4 页，寄存器 0x05 ⁽⁵⁾	NVM 第 4 页，寄存器 0x05 ⁽⁵⁾	0xFF	0xFF	0xFF	0xFF
0xF5	0x00	0x00	不适用	不适用	不适用	不适用
0xF6	0x0A	0x0A	0xFF	0xFF	0xFF	0xFF
0xF7	0x00	0x00	不适用	不适用	不适用	不适用
0xF8	第 4 页，寄存器 0x09 ⁽⁶⁾	NVM 第 4 页，寄存器 0x09 ⁽⁶⁾	0xFF	0xFF	0xFF	0xFF
0xF9	0x00	0x00	不适用	不适用	不适用	不适用
0xFA	0x00	0x00	不适用	不适用	不适用	不适用
0xFB	0x00	0x00	不适用	不适用	不适用	不适用
0xFC	0x00	0x00	不适用	不适用	不适用	不适用
0xFD	0x00	0x00	不适用	不适用	不适用	不适用
0xFE	0x00	0x00	不适用	不适用	不适用	不适用
0xFF	0x00	0x00	不适用	不适用	不适用	不适用

- (1) 第 0 页的寄存器地址 0x01、0x02 和 0x03 (DEV_REV、NVM_CODE_1 和 NVM_CODE_2) 在出厂时进行了编程，不能更改。这些值是寄存器 CRC 计算的一部分，必须从器件读取。
- (2) NVM 表示该寄存器受 NVM 支持，并且将从 GUI 生成的 .BIN 文件中读取值 (请参阅参考文献 3)。
- (3) NA 表示 CRC 掩码不适用。在算法中，用于更新 CRC 的值是数据和掩码的逻辑“与”。如果数据或掩码为“0x00”，则结果为 0x00。根据算法，这可以解释为未定义的寄存器。
- (4) 寄存器地址 0x92、0x9B 和 0xC3 混合了受 NVM 支持的位和不受 NVM 支持的位。受 NVM 支持的位 ESM_MCU_EN 和 ESM_SOC_EN (分别位于寄存器 0x92 和 0x9B 中) 通常为“0”。不建议在上电后立即启用 ESM。寄存器 0xC3 中的 FIRST_STARTUP_DONE 不受 NVM 支持，默认情况下为“0”。
- (5) 第 4 页寄存器 0x05 是 WD_LONGWIN 寄存器。该值取自 GUI 生成的 .BIN 文件中的地址 0x405。
- (6) 第 4 页寄存器 0x09 是 WD_THR_CFG 寄存器。该值取自 GUI 生成的 .BIN 文件中的地址 0x405。只有第 6 位 WD_EN 受 NVM 支持，但默认情况下也会设置其他位。因此，该值为 0xFF 或 0xBF。

D I²C 串行接口示例

表 D-1 中的 TPS6594-Q1 示例取自可扩展 PMIC GUI 实现。在此示例中，目标器件的初始 I²C 地址为 0x48，更新后的 I²C 地址为 0x28。

表 D-1. 更新了 I2C1 地址的 I²C 示例

指令	I ² C 地址 (页)	读/写	寄存器地址	数据	说明
1	0x48 (0)	写入	0xA2	0x00	复位解锁逻辑
2	0x48 (0)	写入	0xA2	0x98	NVM 解锁
3	0x48 (0)	写入	0xA2	0xB8	
4	0x48 (0)	写入	0xA2	0x13	
5	0x48 (0)	写入	0xA2	0x7D	
6	0x48 (0)	读取	0xA3	0xC0	确认 NVM 已成功解锁；已设置第 6 位。
7	0x48 (0)	写入	0xA3	0xC1	暂停 PFSM
8	0x49 (1)	写入	0x18	0x0D	该指令解锁频率选择，以便可以随第 0 页的其他更新一起更改 BUCK 频率 (第 0 页寄存器 0x8A)。更新第 1 页时，该寄存器设置为适当的应用值。
9	0x48 (0)	写入	0x31	0x20	将 GPIO1、GPIO2 和 GPIO3 (LP876x) 更新为所需的最终串行接口设置。
10	0x48 (0)	写入	0x32	0x40	
11	0x48 (0)	写入	0x33	0x10	
12	0x49 (1)	写入	0x22	0x28	将 I2C1 地址更新为 0x28。以下所有寄存器访问都基于该地址。
13	0x29 (1)	写入	0x23	0x12	将 I2C2 地址更新为 0x12
14	0x29 (1)	写入	0x1A	0x00	更新串行接口模式。此时，如果串行接口更改为 SPI 或如果启用了 CRC，则必须进行相关的更改，然后再继续执行下一条指令。
15	0x28 (0)	写入	0x04-0xD1	阵列	根据表 A-1 将内容写入第 0 页寄存器映射。将表 B-1 中的寄存器恢复为默认值。
16	0x29 (1)	写入	0x01-0x43	阵列	根据表 A-1 将内容写入第 1 页寄存器映射。
17	0x12 (4)	写入	0x05、0x09	阵列	将内容写入第 4 页
18	0x28 (0)	写入	0xA4	0x00	将 PFSM 控制设置为子页 0
19	0x2B (3)	写入	0x00-0xFF	阵列	将内容写入第 3 子页 0
20	0x28 (0)	写入	0xA4	0x01	将 PFSM 控制设置为子页 1
21	0x2B (3)	写入	0x00-0xFF	阵列	将内容写入第 3 子页 1
22	0x28 (0)	写入	0xA4	0x02	将 PFSM 控制设置为子页 2
23	0x2B (3)	写入	0x00-0xFF	阵列	将内容写入第 3 子页 2
24	0x28 (0)	写入	0xA4	0x00	将 PFSM 控制设置为子页 0
25	0x28 (0)	写入	0xF0-0xFB	0x00	清除寄存器 CRC 内容
26	0x28 (0)	写入	0xEF	0x02	运行 CRC BIST 并更新寄存器 CRC 值
27	0x28 (0)	读取	0xFB	非零值	这是一项简单的检查，用于查看 CRC 是否已完成。这项检查只是查看该值是否从 0x00 更新 (在第 24 步中清除)。
28	0x28(0)	写入	0xF0-0xF3	阵列	计算 16 位用户寄存器 CRC 值，并更新 REGMAP_USER_INCLUDE_PERSIST_CRC16 和 REGMAP_USER_INCLUDE_EXCLUDE_CRC16 寄存器。
29	0x29 (1)	写入	0xE1	0x00	准备 NVM 以从寄存器映射接收更新。如果希望锁定 EEPROM 以便以后无法进行更新，则在传输命令之前，先将除 0xA5 以外的任何值写入地址 0x141：第 1 页寄存器地址 0x41。
30	0x29 (1)	写入	0xEF	0x02	开始从寄存器映射到 NVM 的传输。
31	0x29 (1)	读取	0xF3	0x04	轮询位 1。当位 1 清除时，传输完成。位 2 可以是“0”或“1”。
32	0x28 (0)	写入	0xA2	0x00	复位解锁逻辑

E 修订历史记录

Changes from Revision * (October 2022) to Revision A (April 2023)	Page
• 更新了整个文档中的表格、图和交叉参考的编号格式.....	1
• 添加了注意说明，以突出 LP8764-Q1 和其他器件之间的差异.....	6
• 添加了注意事项，以突出 LP8764-Q1 和其他器件之间的差异。.....	9
• 进行了更改，以阐明更新寄存器 0x141 时锁定的内容。.....	9
• 从表中删除了要保持不变的地址。.....	14
• 更改了第 1 页更新的起始地址。.....	25

重要声明和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2023，德州仪器 (TI) 公司