Application Note

ZHCAFE7 – May 2025

TPS6594/LP8764 PMIC 应用指南（二）

Lijia Zhu

摘要

随着新能源汽车的高速发展，汽车从电动化转向智能化，高阶辅助驾驶在各大 OEM 和 Tier1 的快速落地，TDA4 正在被广泛的使用在 ADAS 域控制器，车身域控制器和激光雷达等多种终端应用。以 TPS6594/LP8764 为代表的 TI 新一代 PMIC 家族，不仅是 TDA4 SOC 的最佳供电方案，也是其他型号 SOC 供电的良好选择。TPS6594/LP8764 具有高集成度，高可拓展性，支持高级别功能安全等特性，功能多且较复杂，应用起来会有不小的难度。本系列文章会从 TPS6594/LP8764 PMIC 的芯片主要机制介绍、系统设计注意点、常见问题定位思路、自定义 PMIC 固件（NVM）等方面进行分享。本文是系列文章的第二篇，接上一篇继续对 PMIC 芯片内部的主要机制进行介绍。

目录

1 引言 3
2 主要模块简介 3
2.1 NVM 和 register 4
2.1.1 Register 分页 4
2.1.2 NVM 和 register 空间 4
2.2 有限状态机 FSM 5
2.2.1 FFSM 5
2.2.2 中断处理机制 8
2.2.3 PFSM 13
2.3 SPMI 通信机制 17
3 参考文献 18

图

Figure 1. TPS6594 数字逻辑模块架构简图 3
Figure 2. TPS6594 NVM 和寄存器关系 5
Figure 3. TPS6594 FFSM 架构 6
Figure 4. TPS6594 VMON OV/UV BIST 示意图 7
Figure 5. TPS6594 中断控制架构 8
Figure 6. 中断信息汇总 1 10
Figure 7. 中断信息汇总 2 11
Figure 8. FSM trigger 汇总 11
Figure 9. TPS6594 中断寄存器汇总 12
Figure 10. PDN 0C FSM 架构 14
Figure 11. FSM 局部 1 15
Figure 12. FSM 局部 2 15

Figure 13. FSM 局部 3 16
Figure 14. FSM trigger 优先级 16
Figure 15. SPMI 框图 17
Figure 16. SPMI BIST 波形 18

1 引言

TPS6594/LP8764 系列是 TI 新一代 PMIC 产品，它具有高集成度，高可拓展性和高功能安全等特点。TPS6594 内部集成了 5 个 BUCK convertor 和 4 个 LDO，并具有丰富的 GPIO，这些 GPIO 可以通过 NVM 配置为多种功能，包括外部唤醒输入，看门狗喂狗输入，使能输入，外部电压监控输入，外部高边开关/LDO/BUCK 的使能控制等功能，进一步提高了器件的拓展性。LP8764 芯片和 TPS6594 芯片具有类似的芯片架构，常常作为辅 PMIC 和 TPS6594 在系统中配合使用。TPS6594/LP8764 使用 SPMI 接口进行多个 PMIC 间的状态同步，多个并联工作的 PMIC 的状态机可以视为一个整体，在功能安全方面，TPS6594/LP8764 在设计阶段遵照 ISO26262 的要求进行开发，器件设计有大量功能安全的检测机制，支持 ASILD 级别的功能安全。

2 主要模块简介

TPS6594 和 LP8764 功能模块类似，这里以 TPS6594 为例对芯片内部主要模块进行介绍。我们可以粗略的把 TPS6594 分成以下几个模块：

1. VSYS 输入控制模块
2. 电源分配模块
3. 电压监控模块
4. BUCK 模块
5. LDO 模块
6. IO 模块
7. 数字逻辑模块

前文对 1-6 的模块进行了介绍，本文将对数字逻辑模块进行介绍。数字逻辑模块是整个 PMIC 的控制核心，其架构简图如图 1 所示，包括存储 PMIC 运行逻辑的 NVM（EEPROM）和记录状态 register 空间、控制运行逻辑的 FSM 模块、确保 SOC 正常运行的看门狗和 ESM 机制、多片协同的 SMPI 机制等主要模块，本文将会集中在这些主要模块进行介绍。

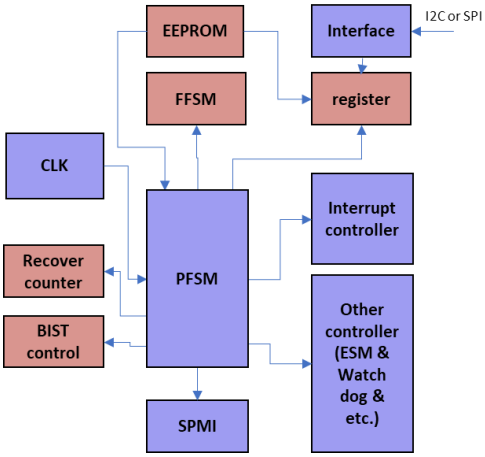

Figure 1. TPS6594 数字逻辑模块架构简图

2.1 NVM 和 register

2.1.1 Register 分页

TPS6594 的寄存器分为若干页，每一页的寄存器具有不同的功能：

Page 0: User Registers，包括开放给客户的寄存器，包括对各个模块（BUCK，LDO，IO 等）的设置和状态

Page 1: NVM Control, Configuration, and Test Registers，包括 NVM 配置的寄存器和测试模式的寄存器

Page 2: Trim Registers，此页不可通过通信接口改写

Page 3: SRAM for PFSM Registers，是 PFSM 的运行代码

Page 4: Watchdog Registers，看门狗相关的寄存器，开放给客户访问

外部 MCU 可以使用 SPI 或 I2C 访问 TPS6594 的寄存器，两种模式的切换由 NVM 的 SERIAL_IF_CONFIG 决定。

在 I2C 模式下，芯片有两个 I2C 接口，可以使用 I2C1 访问 page0-3 的寄存器，使用的 PMIC 预设的 I2C1 地址访问 page0，它由 NVM 中 I2C1_ID_REG 寄存器设置，要访问 page1 寄存器，则使用 I2C1 地址+1 的地址，以此类推。Page4 比较特殊，它有独立的 I2C2 地址，由 I2C2_ID_REG 寄存器设置。当 I2C2 看门狗喂狗模式未使能时，I2C1 占用 5 个地址（I2C1_ID，I2C1_ID+1，I2C1_ID+2，I2C1_ID+3，I2C2_ID）,可以访问全部的五页寄存器，当使能 I2C2 看门狗喂狗时，I2C1 占用 4 个地址（I2C1_ID，I2C1_ID+1，I2C1_ID+2，I2C1_ID+3）访问 page0-3，I2C2 占用 1 个地址（I2C2_ID）访问 page4。

SPI 接口模式下，page 信息通过 SPI 数据包中特定的 page 字段区分，具体参考手册相关内容。

2.1.2 NVM 和 register 空间

PMIC 内部有两个相对独立的存储空间，即 NVM 空间和寄存器空间，NVM 空间的数据掉电不丢失，它存储了器件的上电默认工作状态，寄存器空间的数据随 VCCA 的掉电而丢失。NVM 和寄存器空间之间可以互相交互，在器件上电过程中，NVM 的内容被加载到寄存器空间，后续器件按照寄存器空间的内容运行；反之也可以将寄存器空间的数据写入 NVM，这个就是通过外部接口更新 NVM 的途径。需要额外注意，访问寄存器数据要求 VIO 供电存在，某些 PDN（电源设计）的 VIO 是从 PMIC 的输出取得的，使用这些 PDN 的板子在故障场景时因为 VIO 无供电，无法访问寄存器，需要外部提供 VIO 供电。

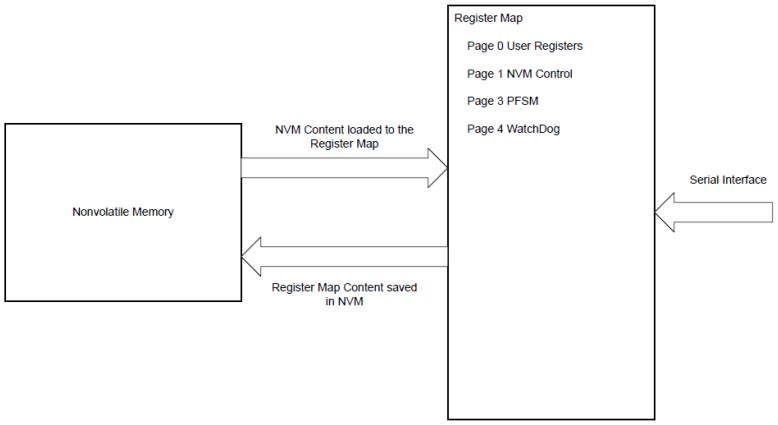

Figure 2. TPS6594 NVM 和寄存器关系

2.2 有限状态机 FSM

TPS6594 内部集成了有限状态机（Finite State Machine，FSM），它通过切换不同的状态控制了 PMIC 的行为逻辑。它包括三部分：

硬件固定的 FSM 部分被称为 FFSM（Fixed Finite State Machine），它由硬件固定，控制的逻辑包括那些所有应用场景都通用的固定功能。

由 NVM 设定的可编程 FSM 部分被称为 PFSM（Pre-configurable Finite State Machine），其逻辑由 NVM page3 的内容决定。PFSM 与 FSM 协同工作,以充分利用这两者各自的优势，显著降低完全可配置的 FSM 的高复杂度，同时提供足够的灵活性可以更具不同的应用场景变更状态机逻辑。

中断处理机制控制中断的处理、上报以及和 FSM 的交互。中断处理机制通过 MASK 机制对不同的中断进行处理，判断是否上报 MCU（拉低中断管脚）和是否影响 FSM。

2.2.1 FFSM

FFSM 控制 PMIC 电源轨启动前和故障后的器件工作逻辑。需要注意，在 PMIC 处于 FFSM 的任何一个状态，ENABLE_DRV 始终为低电平。

FFSM 状态机如图 3 所示，共有如下几个状态：

1. NO SUPPLY：器件处于关机状态，VCCA>UV 进入 INIT 状态
2. BACKUP：又叫 RTC backup battery，此状态下只有 RTC 时钟还在工作。任何其他状态下，满足 VCCA<UVLO 即可进入此状态。此状态下若 VRTC 也欠压，则进入 NO SUPPLY 状态。可以通过直接断开 VCCA 的方式（shelf mode）避免进入此状态而直接进入 NO SUPPLY 状态。
3. LP_STANDBY：低功耗休眠模式，此时只有 RTC 域工作，可以通过 RTC 计时器或者 LPWKUP 管脚来唤醒。在 mission state（PFSM 部分）中，收到关机请求（off request）中断信号且 LP_STANDBY_SEL 为 1 时，进入此状态。若 LP_STANDBY_SEL 为 0，即使有关机请求，PFSM 仅进入 STANDBY 状态。

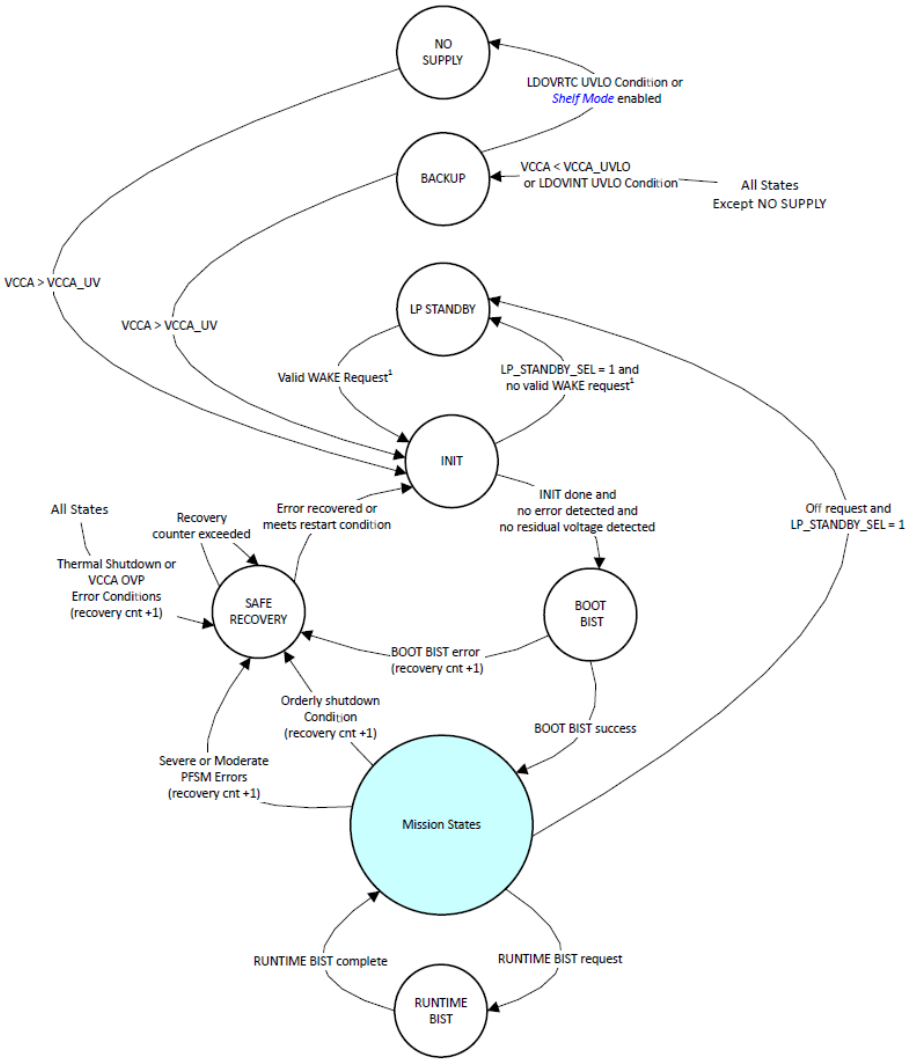

Figure 3. TPS6594 FFSM 架构

4. INIT：初始化状态，进入条件是 VCCA ≥ VCCA_UV，或从 LP_STANDBY 唤醒，NVM 在此阶段 load 到寄存器空间中。

5. BOOT BIST：启动阶段的自检阶段，INIT 阶段后自动进入，包括 LBIST 和 ABIST 两部分。LBIST 包括对内部数字模块的自检（比如 PFSM，WD，clock 等），ABIST 包括对各电源轨的 VMON OV/UV 的上电自检和过温监控模块的自检。LBIST 和 ABIST 是同步进行的，没有先后顺序，因此可能会出现 BIST 即有 BIST PASS 又有 BIST FAIL 中断上报的情况。当 BIST 失败，系统会进入 safe RECOVERY 状态，并重新开始 BOOT BIST，从 SAFE RECOVERY 状态回到 BOOT BIST 的场景，仅进行 ABIST。部分寄存器可以用来关闭部分的 BOOT BIST 的功能，具体如下：

    1. FAST_BOOT_BIST 置 1 可以跳过从 NO SUPPLY 进入 BOOT BIST 的 LBIST 流程来加速启动流程。

2. FAST_BIST 置 1 可以跳过从 LP_STANDBY 进入 BOOT BIST 的 LBIST 流程来加速唤醒过程。

3. REG_CRC_EN 置 0 以禁用寄存器的 CRC 校验功能。

4. VMON_ABIST_EN 置 0 禁用的 VMON OV/UV 的 BOOT BIST。

这里额外提一句 VMON OV/UV 的 BIST，如图 4 左右两侧所示分别为 BUCK 和 LDO 的 VMON OV/UV BIST 示意图，在每个电源轨启动后，经过确定的延时（startup delay+ ramp up time+ setting time）对此电源轨对应的 VMON OV/UV 的 BIST 会开启。

在蓝色虚线框所示的 50us 时间内，OV/UV 对状态机屏蔽输出结果，同时在前 25us 会交换 OV 和 UV 的门限，此时 BIST 逻辑预期看到 OV 和 UV 都被触发，后 25us 则是换回正常工作的设置，BIST 逻辑预期看到 OV 和 UV 都不触发。很显然，在这 50us 内电源轨必须达到预期的电压值，否则 BIST 就会失败，此时中断寄存器会上报 BIST FAIL。一般这种情况是电源轨在上电过程中过缓或者输出电压和预期电压不一致导致的，需要注意这种情况下 PMIC 不会上报 UV 或 OV 的告警。

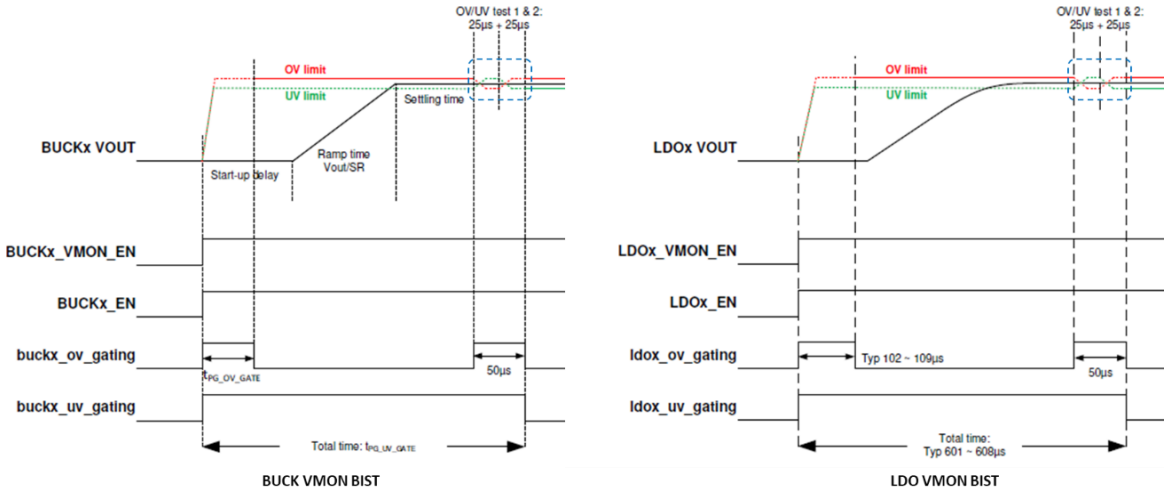

Figure 4. TPS6594 VMON OV/UV BIST 示意图

6. RUNTIME BIST：在器件运行过程中由 MCU 主动发起的 BIST 操作，若 BIST 通过，系统继续运行，BIST PASS 中断置位。若 BIST 失败，系统进入 SAFE RECOVERY 状态，BIST FAIL 中断置位。

7. SAFE RECOVERY：系统出现故障时进入此状态，所有输出关闭。当故障在 recovery time（不同故障此时延不同）内消失，且 recovery counter 计数器没有超过门限时，状态机会进入 INIT 状态尝试重新启动并把 recovery counter 计数器+1。当 recovery counter 达到上限后，必须重启芯片（VCCA <UVLO）PMIC 才会重新输出。Recovery counter 计数器结果在 0x83 寄存器 bit0-3，recovery counter 上限（RECOV_CNT_THR）由 0x84 寄存器 bit0-3 设置。

在系统运行过程中，若偶发进入 SAFE RECOVERY，或是通过写 SOFT REBOOT 手动重启 PMIC 会导致 recovery counter 积累一定的计数，MCU 可以在到达上限之前清零 recover counter 以避免长时间运行次数累计而计数器达到上限，当 recover counter 达到上限之后，即使清零计数器，PMIC 也不会重新启动，必须重启芯片。

一些特殊的系统要求（比如 debug）PMIC 在故障场景能够无限次重启而不是到达计数器上限后停留在 SAFE RECOVER 状态，可以通过在 NVM 中设置 RECOV_CNT_THR 到 0x0 来实现，需要注意运行过程中修改 RECOV_CNT_THR 寄存器为零不会达到同样的效果，因为从 SAFE RECOVERY 恢复经过 INIT 状态时，NVM 里的 RECOV_CNT_THR 会覆盖寄存器里面的 0 值。

8. Mission State：准确来讲它不是一个状态，而是嵌套在 FFSM 里面的一个 PFSM，PMIC 经过 BIST 之后会进入 PFSM 部分。

2.2.2 中断处理机制

进入 PFSM 部分之前，必须先理解中断处理机制。TPS6594 具有可分层屏蔽的中断机制，事件（中断源）的作用结果有两个，一是上报外部 MCU（拉低中断管腿），二是触发 FSM 的状态切换。如图 5 所示是手册中的示意图，最上层是所有故障/事件，经过第一层中断屏蔽逻辑后，未屏蔽的中断会输出到中断管脚上，同时送到 PFSM 所在的下一层逻辑。而各个电源轨会根据 power group 的设定，把各自对应故障送到对应的 power group 汇总，power group 的汇总结果汇报给 FSM。

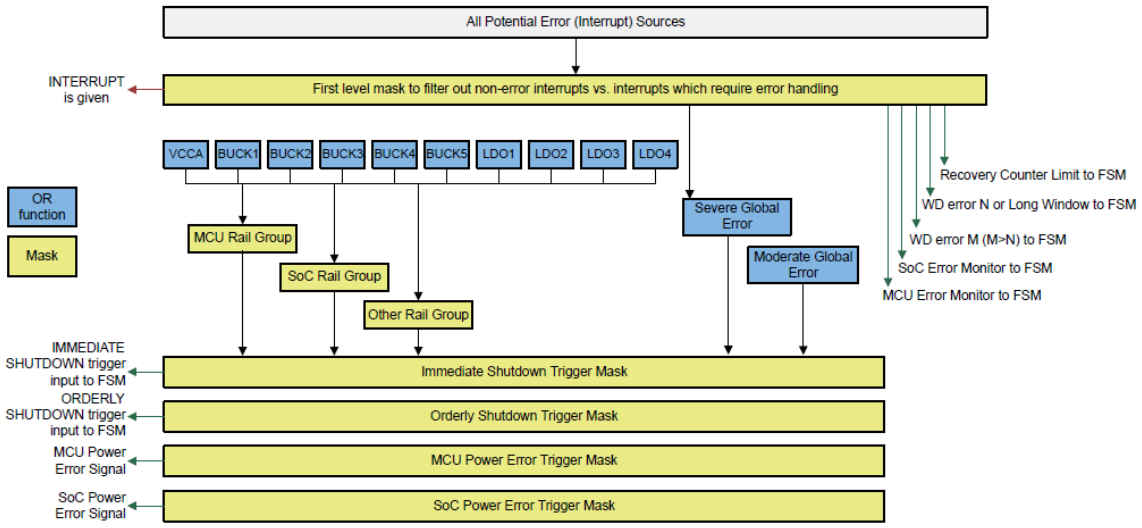

Figure 5. TPS6594 中断控制架构

图 5 相对而言还是比较难理解，尤其这里的 MASK 机制，混淆了上报 MCU 的 MASK 和作为 FSM trigger 的 MASK。要更深入理解中断机制，可以参考手册的 Summary of Interrupt Signals 表格，图 6 和 7 是截取了它的一部分作为例子，把这张表分成三部分：

1. 最左边列举了各种可能出现的事件，它可以是外部信号或内部故障，比如使能拉起和 SPMI 通信异常，它也可以是直接的信号或者是汇总而来的二级中断，比如 BUCK 欠压和 MCU power group error。

2. 中间部分是事件对 FSM 的作用，其中第一列是对应的 FSM trigger 名称以及是否可屏蔽（若可屏蔽则提供屏蔽寄存器），第二列是事件对 FSM 的影响，第三列是因为此事件 FSM 转移后要恢复到原先状态的条件。

3. 最右边部分是事件对中断管脚的作用，其中第一列是事件对应的中断寄存器（事件消失后仍锁存此状态），第二列是提供了屏蔽此中断上报的寄存器，第三列是事件对应的实时状态寄存器，第四列是清除中断寄存器的方法。

从图 6 的中断和 FSM 相互作用部分的例子可以看到，BUCK 和 LDO 对 FSM 的作用要经过 power group 这一环节，这对应了图 5 左下方的部分。

从图 7 可以看到，除了 BUCK 和 LDO 外有大量事件直接作为 FSM 的 trigger。所有的 FSM trigger 可以在手册的 PFSM Trigger Selections 表格找到完整的说明，截取部分如图 8 所示。

EVENT	TRIGGER FOR FSM	RESULT (1)	RECOVERY	INTERRUPT BIT	MASK FOR INTERRUPT	LIVE STATUS BIT	INTERRUPT CLEAR
BUCK regulator forward current limit triggered	**EN_ILIM_FSM_CTRL=1:** According to BUCKn_GRP_SEL and x_RAIL_TRIG bits **EN_ILIM_FSM_CTRL=0:** N/A	**EN_ILIM_FSM_CTRL=1:** Transition according to FSM trigger and interrupt **EN_ILIM_FSM_CTRL=0:** Interrupt only	Depends on PFSM configuration, see PFSM transition diagram	BUCKn_ILIM_INT = 1	BUCKn_ILIM_MASK	BUCKn_ILIM_STAT	Write 1 to BUCKn_ILIM_INT bit Interrupt is not cleared if current limit violation is active
LDO regulator current limit triggered	**EN_ILIM_FSM_CTRL=1:** According to LDOn_GRP_SEL and x_RAIL_TRIG bits **EN_ILIM_FSM_CTRL=0:** N/A	**EN_ILIM_FSM_CTRL=1:** Transition according to FSM trigger and interrupt **EN_ILIM_FSM_CTRL=0:** Interrupt only	Depends on PFSM configuration, see PFSM transition diagram	LDOn_ILIM_INT = 1	LDOn_ILIM_MASK	LDOn_ILIM_STAT	Write 1 to LDOn_ILIM_INT bit Interrupt is not cleared if current limit violation is active
BUCK output or switch short circuit detected	According to BUCKn_GRP_SEL and x_RAIL_TRIG bits	Regulator disable and transition according to FSM trigger and interrupt	Depends on PFSM configuration, see PFSM transition diagram	BUCKn_SC_INT = 1	N/A	N/A	Write 1 to BUCKn_SC_INT bit
LDO output short circuit detected	According to LDOn_GRP_SEL and x_RAIL_TRIG bits	Regulator disable and transition according to FSM trigger and interrupt	Depends on PFSM configuration, see PFSM transition diagram	LDOn_SC_INT = 1	N/A	N/A	Write 1 to LDOn_SC_INT bit
BUCK output residual voltage violation	**BUCKn_RV_SEL = 1** According to BUCKn_GRP_SEL and x_RAIL_TRIG bits **BUCKn_RV_SEL = 0** N/A	**BUCKn_RV_SEL = 1** Regulator disable and transition according to FSM trigger and interrupt **BUCKn_RV_SEL = 0** N/A	Depends on PFSM configuration, see PFSM transition diagram	BUCKn_SC_INT = 1	N/A	N/A	Write 1 to BUCKn_SC_INT bit
LDO output residual voltage violation	**LDOn_RV_SEL = 1** According to LDOn_GRP_SEL and x_RAIL_TRIG bits **LDOn_RV_SEL = 0** N/A	**LDOn_RV_SEL = 1** Regulator disable and transition according to FSM trigger and interrupt **LDOn_RV_SEL = 0** N/A	Depends on PFSM configuration, see PFSM transition diagram	LDOn_SC_INT = 1	N/A	N/A	Write 1 to LDOn_SC_INT bit
BUCK regulator overvoltage	According to BUCKn_GRP_SEL and x_RAIL_TRIG bits	Transition according to FSM trigger and interrupt	Depends on PFSM configuration, see PFSM transition diagram	BUCKn_OV_INT = 1	BUCKn_OV_MASK	BUCKn_OV_STAT	Write 1 to BUCKn_OV_INT bit Interrupt is not cleared if it is active
BUCK regulator undervoltage	According to BUCKn_GRP_SEL and x_RAIL_TRIG bits	Transition according to FSM trigger and interrupt	Depends on PFSM configuration, see PFSM transition diagram	BUCKn_UV_INT = 1	BUCKn_UV_MASK	BUCKn_UV_STAT	Write 1 to BUCKn_UV_INT bit Interrupt is not cleared if it is active
LDO regulator overvoltage	According to LDOn_GRP_SEL and x_RAIL_TRIG bits	Transition according to FSM trigger and interrupt	Depends on PFSM configuration, see PFSM transition diagram	LDOn_OV_INT = 1	LDOn_OV_MASK	LDOn_OV_STAT	Write 1 to LDOn_OV_INT bit Interrupt is not cleared if it is active
LDO regulator undervoltage	According to LDOn_GRP_SEL and x_RAIL_TRIG bits	Transition according to FSM trigger and interrupt	Depends on PFSM configuration, see PFSM transition diagram	LDOn_UV_INT = 1	LDOn_UV_MASK	LDOn_UV_STAT	Write 1 to LDOn_UV_INT bit Interrupt is not cleared if it is active

Figure 6.　中断信息汇总 1

EVENT	TRIGGER FOR FSM	RESULT (1)	RECOVERY	INTERRUPT BIT	MASK FOR INTERRUPT	LIVE STATUS BIT	INTERRUPT CLEAR
Thermal shutdown, orderly sequenced	ORDERLY_SHUTDOWN (MODERATE_ERR_INT)	All regulators disabled and Output GPIOx set to low in a sequence and interrupt(1)	Automatic start-up to STARTUP_DEST[1:0] state after temperature is below TWARN level	TSD_ORD_INT = 1	N/A	TSD_ORD_STAT	Write 1 to TSD_ORD_INT bit Interrupt is not cleared if temperature is above thermal shutdown level
Thermal shutdown, immediate	IMMEDIATE_SHUTDOWN (SEVERE_ERR_INT)	All regulators disabled with pull-down resistors and Output GPIOx set to low immediately and interrupt(1)	Automatic start-up to STARTUP_DEST[1:0] state after temperature is below TWARN level	TSD_IMM_INT = 1	N/A	TSD_IMM_STAT	Write 1 to TSD_IMM_INT bit Interrupt is not cleared if temperature is above thermal shutdown level
BIST error	ORDERLY_SHUTDOWN (MODERATE_ERR_INT)	All regulators disabled and Output GPIOx set to low immediately and interrupt(1)	Automatic start-up to STARTUP_DEST[1:0] state	BIST_FAIL_INT = 1	BIST_FAIL_MASK	N/A	Write 1 to BIST_FAIL_INT bit
Register CRC error	ORDERLY_SHUTDOWN (MODERATE_ERR_INT)	All regulators disabled and Output GPIOx set to low immediately and interrupt(1)	Automatic start-up to STARTUP_DEST[1:0] state	REG_CRC_ERR_INT = 1	REG_CRC_ERR_MASK	N/A	Write 1 to REG_CRC_ERR_INT bit
SPMI communication error	ORDERLY_SHUTDOWN (MODERATE_ERR_INT)	All regulators disabled and Output GPIOx set to low immediately and interrupt(1)	Automatic start-up to STARTUP_DEST[1:0] state	SPMI_ERR_INT = 1	SPMI_ERR_MASK	N/A	Write 1 to SPMI_ERR_INT bit
SPI frame error	N/A	Interrupt only	Not valid	COMM_FRM_ERR_INT = 1(4)	COMM_FRM_ERR_MASK	N/A	Write 1 to COMM_FRM_ERR_INT bit
I2C1 or SPI CRC error	N/A	Interrupt only	Not valid	COMM_CRC_ERR_INT = 1	COMM_CRC_ERR_MASK	N/A	Write 1 to COMM_CRC_ERR_INT bit
I2C1 or SPI address error(5)	N/A	Interrupt only	Not valid	COMM_ADR_ERR_INT = 1	COMM_ADR_ERR_MASK	N/A	Write 1 to COMM_ADR_ERR_INT bit
I2C2 CRC error	N/A	Interrupt only	Not valid	I2C2_CRC_ERR_INT = 1	I2C2_CRC_ERR_MASK	N/A	Write 1 to I2C2_CRC_ERR_INT bit
I2C2 address error(5)	N/A	Interrupt only	Not valid	I2C2_ADR_ERR_INT = 1	I2C2_ADR_ERR_MASK	N/A	Write 1 to I2C2_ADR_ERR_INT bit

Figure 7.　中断信息汇总 2

Trigger Name	Trigger Source
IMMEDIATE_SHUTDOWN	An error event causes one of the triggers defined in the FSM_TRIG_SEL_1/2 register to activate, and the intended action for the activated trigger is to *immediate shutdown* the device
MCU_POWER_ERROR	Output failure detection from a regulator which is assigned to the MCU rail group (x_GRP_SEL = '01')
ORDERLY_SHUTDOWN	An event which causes MODERATE_ERR_INT = '1'
FORCE_STANDBY	nPWRON long-press event when NPOWRON_SEL = '01', or ENABLE = '0' when NPOWERON_SEL = '00'
SPMI_WD_BIST_DONE	Completion of SPMI WatchDog BIST
ESM_MCU_ERROR	An event which causes ESM_MCU_FAIL_INT
WD_ERROR	An event which causes WD_INT
SOC_POWER_ERROR	Output failure detection from a regulator which is assigned to the SOC rail group (x_GRP_SEL = '10')
ESM_SOC_ERROR	An event which causes ESM_SOC_FAIL_INT
A	NSLEEP2 and NSLEEP1 = '11'. More information regarding the NSLEEP1 and NSLEEP2 functions can be found under Section 8.4.1.2.4.3
WKUP1	A rising or falling edge detection on a GPIO pin which is configured as WKUP1 or LP_WKUP1
SU_ACTIVE	A valid On-Request detection when STARTUP_DEST = '11'
B	NSLEEP2 and NSLEEP1 = '10'. More information regarding the NSLEEP1 and NSLEEP2 functions can be found under Section 8.4.1.2.4.3
WKUP2	A rising or falling edge detection on a GPIO pin which is configured as WKUP2 or LP_WKUP2
SU_MCU_ONLY	A valid On-Request detection when STARTUP_DEST = '10'
C	NSLEEP2 and NSLEEP1 = '01', More information regarding the NSLEEP1 and NSLEEP2 functions can be found under Section 8.4.1.2.4.3
D	NSLEEP2 and NSLEEP1 = '00'. More information regarding the NSLEEP1 and NSLEEP2 functions can be found under Section 8.4.1.2.4.3

Figure 8.　FSM trigger 汇总

TPS6594 的中断告警寄存器也具有分层机制，如图 9 所示，5A 寄存器汇总了全部的中断告警，每一个 bit 的置位代表了对应的下层寄存器有中断，以此类推最多嵌套至三层。

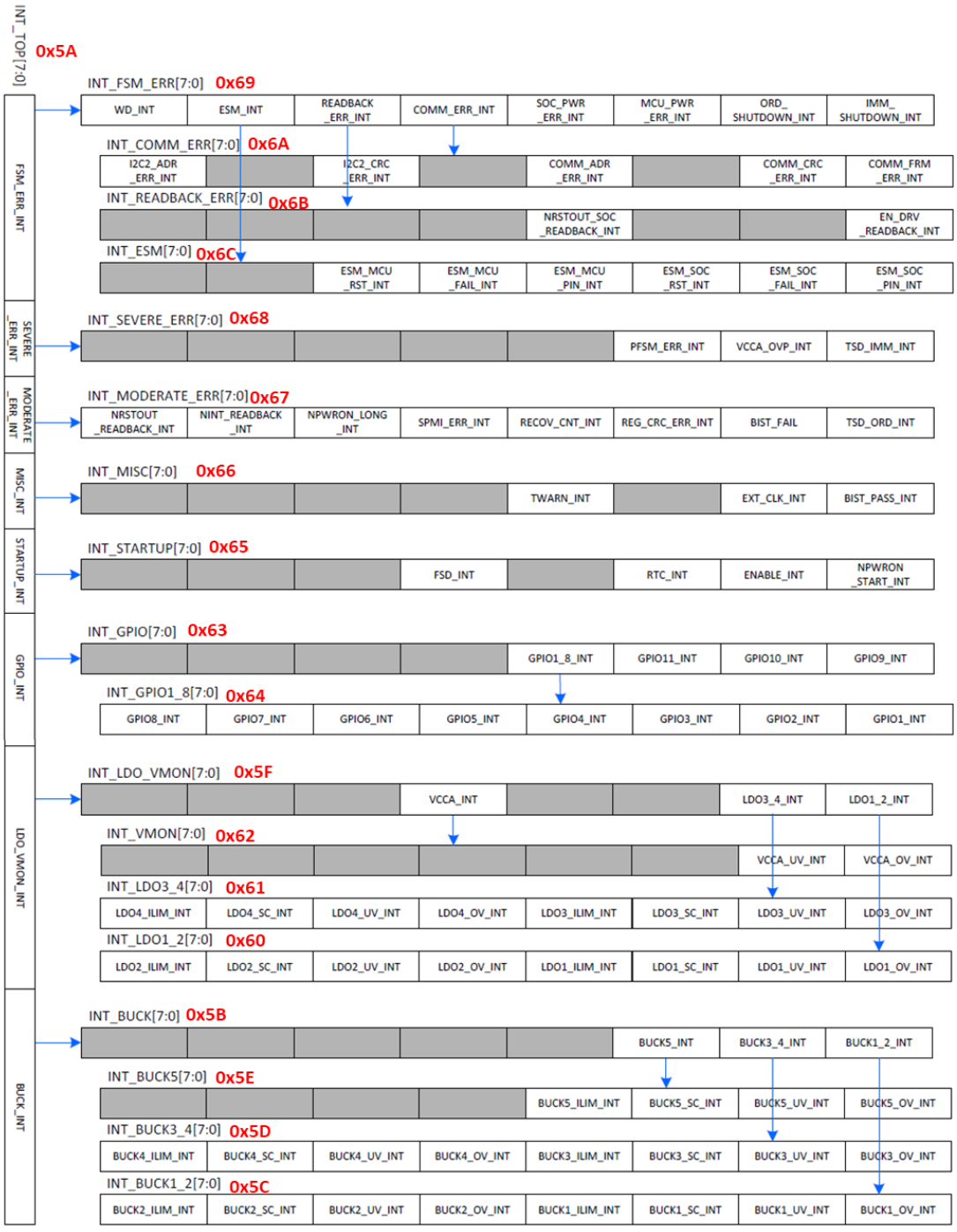

Figure 9. TPS6594 中断寄存器汇总

2.2.3 PFSM

PFSM 控制了器件具体的上下电时序，和其他 page 页的寄存器不同，PFSM 无法在运行过程中修改。需要注意器件手册此章节画的 PFSM 架构仅仅是一个例子，因为 PFSM 的可编辑特性，每个 PMIC 的 PFSM 都不一样，这里以 PDN 0C(TPS65941213+TPS65941111)为例，图 10 所示的是它的整个 FSM 架构，此 PDN 的说明文档，可以在 TI 官网搜索“PDN 0A”关键字得到，其他 PDN 也可以搜索对应的名字获得。

基于这个 PFSM，我们可以来解释一个常见的问题，即 PMIC 正常启动后，若清除 ENABLE INIT 中断，PMIC 会直接掉电：

1. 如图 11 所示，从 standby 到 active 状态的触发条件有两个，SU ACTIVE 和 WKUP1，我们关注 SU ACTIVE 这个 trigger，根据图 8 的信息，它的意思是使能管脚为高，因此外部使能管腿拉高使 PMIC 进入 active 状态，各电源轨对外输出。

2. 在启动后因为 BIST PASS 和 ENABLE INIT 两个中断的存在，中断管腿处于低电平状态，需要清除所有中断寄存器以释放中断管脚，后续故障发生时中断管脚再次拉低，从而 MCU 可以根据此信号对故障及时处理。

3. 如果此时清除 ENABLE INIT 寄存器，刚好满足了 FROCE STANDBY 的条件，即 ENABLE=0，如图 12 所示，PFSM 进入 STANDBY 状态，各电源轨关闭。

4. 如图 13 所示，还有几个 trigger 可以使 FSM 保持于 ACTIVE 状态，这里关注 A 这个 trigger，根据图 8，A 的意思是 NSLEEP bit 为 11。如图 14 所示使各 trigger 的优先级排序，数字越大优先级越高，因此 A trigger 触发 PFSM 停留在 ACTIVE 的优先级高于 FROCE STANDBY 触发 PFSM 的进入 STANDBY 状态的优先级。

5. 因此通过 I2C 寄存器把 NSLEEP0 和 NSLEEP1 都置 1，就会触发 A trigger 使 PFSM 停留在 ACTIVE 状态，此时无论是清除 ENABLE INIT 或者拉低 ENABLE 管腿，PFSM 都会停留在 ACTIVE 状态，所有输出保持不变，不会关机。

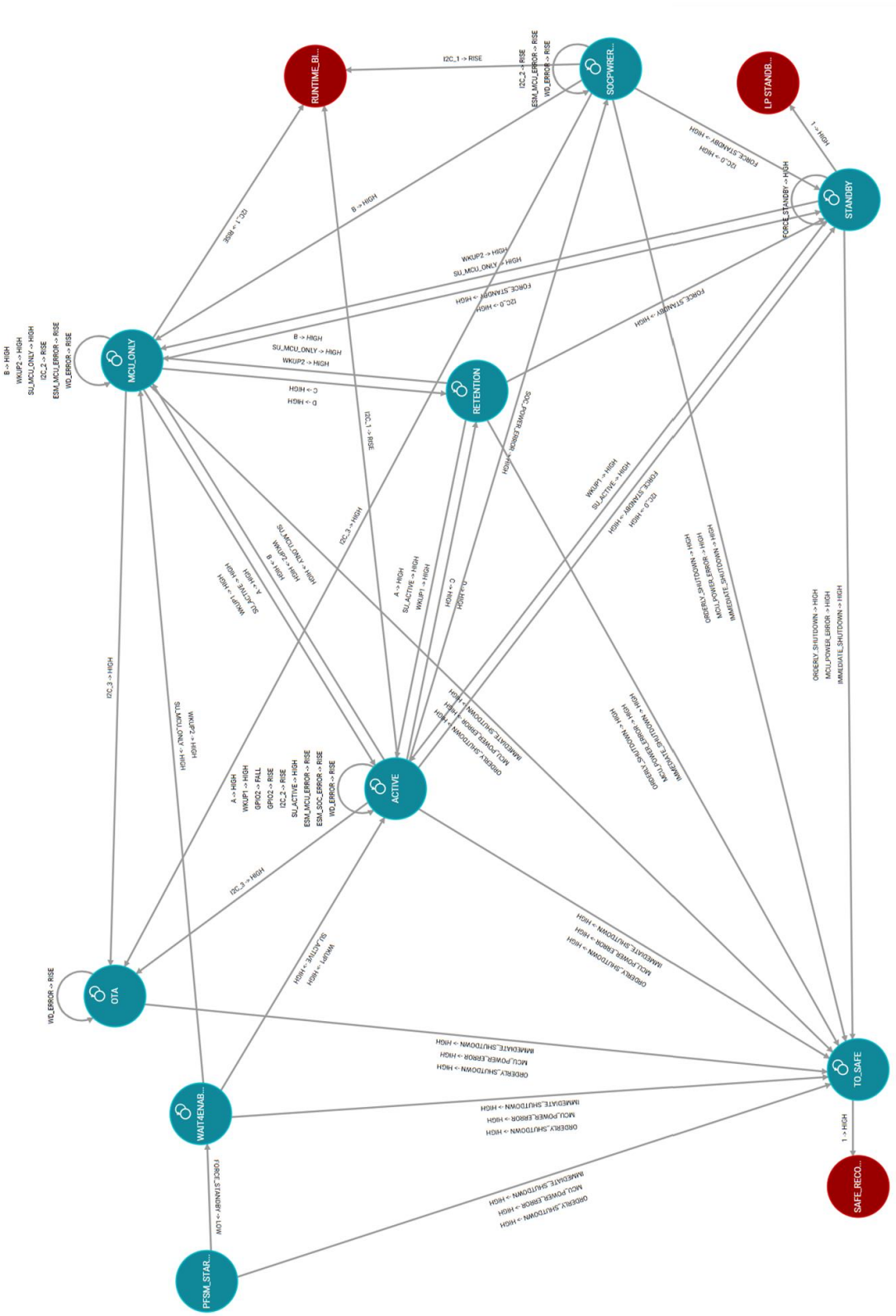

Figure 10. PDN 0C FSM 架构

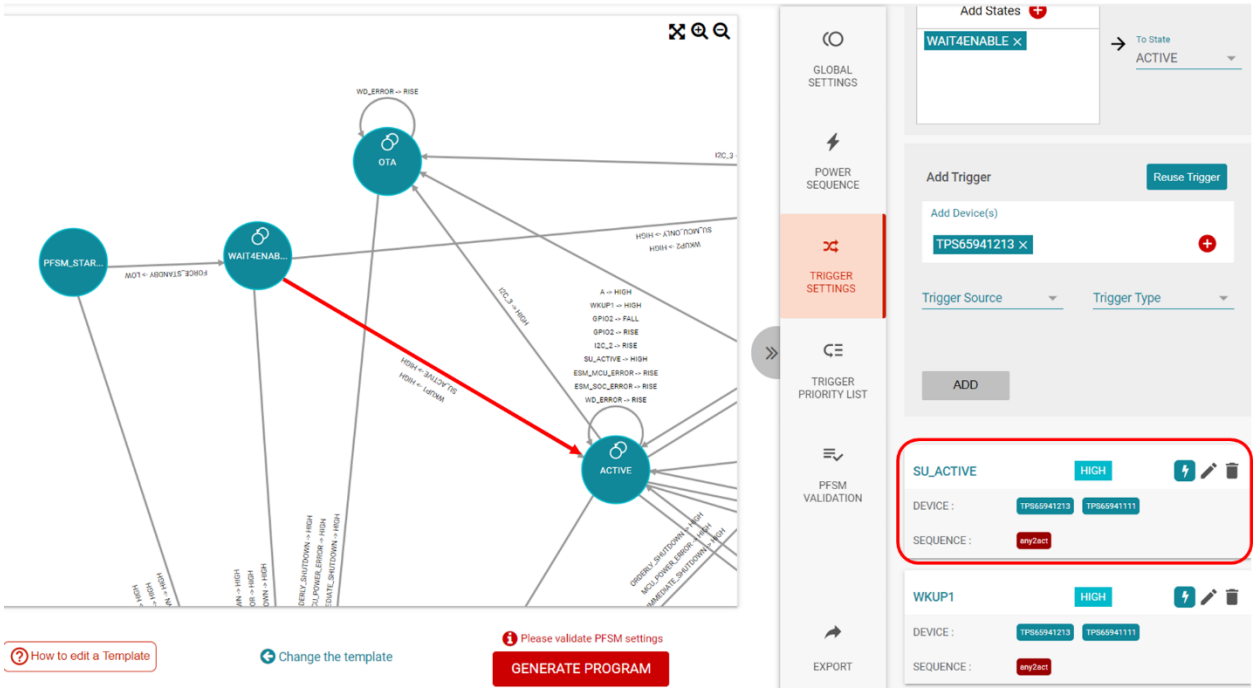

Figure 11. FSM 局部 1

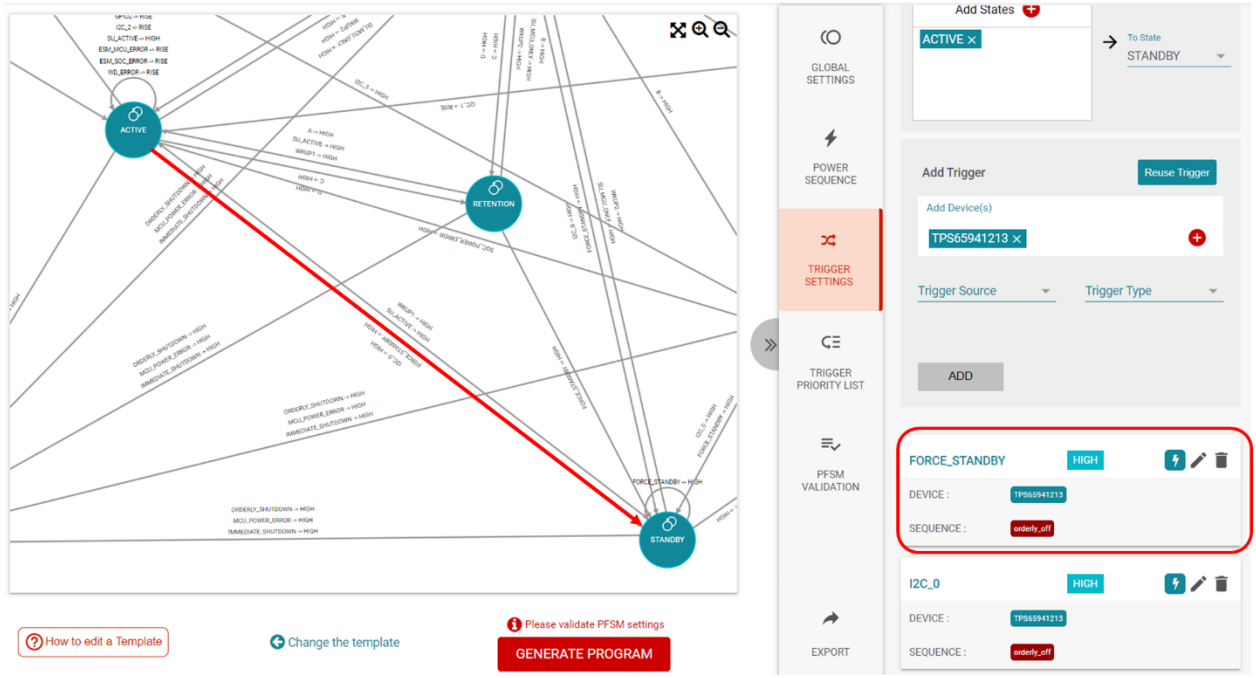

Figure 12. FSM 局部 2

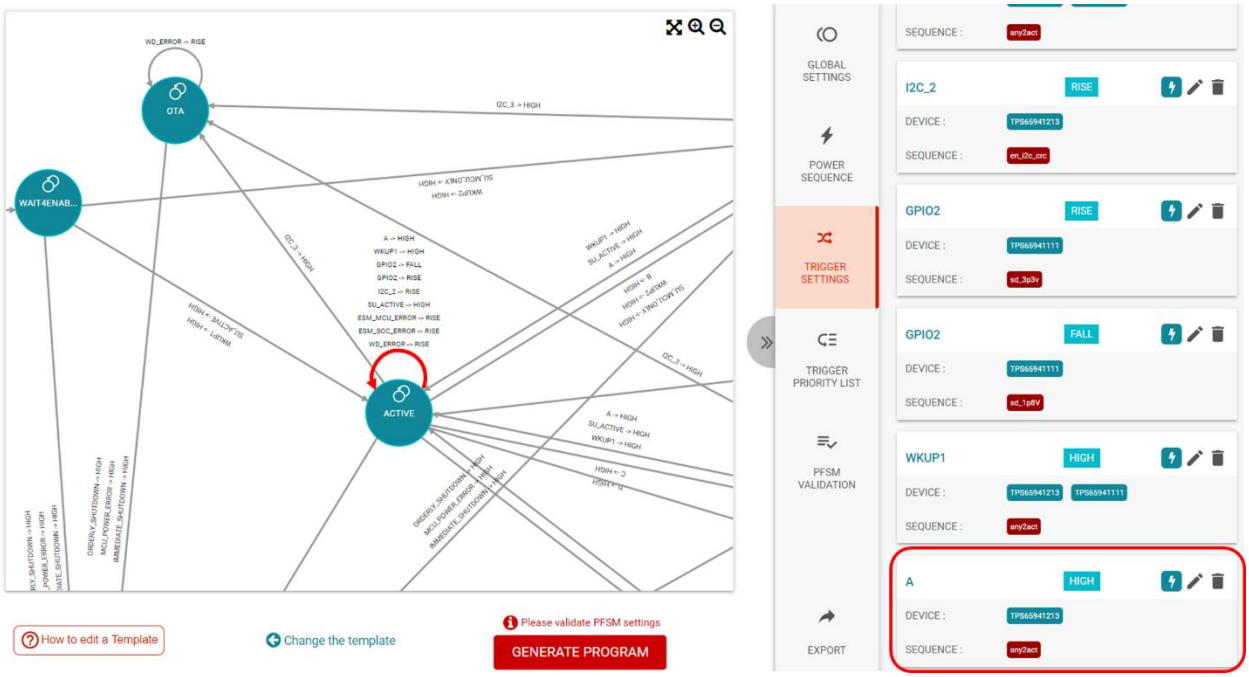

Figure 13. FSM 局部 3

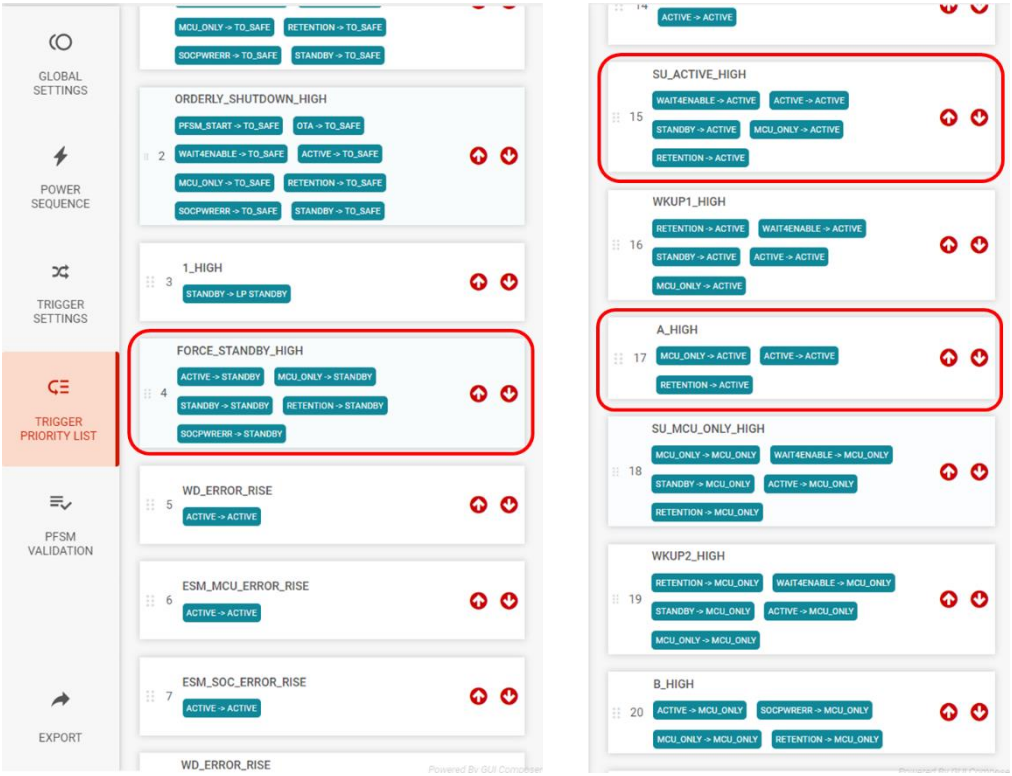

Figure 14. FSM trigger 优先级

2.3 SPMI 通信机制

SPMI（System Power Management Interface）是 MIPI 联盟规定的用在 PMIC 上的通信协议，类似 I2C，SPMI 使用双线进行通信，一根 clock 线一根 data 线，并允许在一组总线上连接多个 PMIC 设备。TPS6594/LP8764 的 SPMI 模块如图 15 所示，具有 controller 和 target 两个部分，当该 PMIC 作为主 PMIC 时，他的 controller 和 target 模块都使能，系统中有且仅有一个主 PMIC。Controller 模块作为总线的控制器发起 SPMI 通信，辅 PMIC 仅使能 target 模块以应答来自 controller 的命令并上报该 PMIC 的状态。

类似 I2C，SPMI 也具有地址机制，分为控制器 ID（CID）和目标 ID（TID）和组 ID（GTID），系统中 Controller 具有唯一的 CID 1，target 模块应答 controller 的数据包使用 CID。TPS6594 最多支持 6 个 TID（主 PMIC TID 为 0101，从第一颗到第五颗辅 PMIC 依次使用 0011，1100，1001，0110 和 1010），一个系统中至多有 6 个 PMIC，当 controller 和特定 target 通信时，数据包使用对应的 TID。所有的 Target 模块监听 GTID 1111 的信息，当 controller 需要广播信息时，数据包使用 GTID。

主 PMIC 的 NVM 中使用了 5 个寄存器依次记录了系统中使用的辅 PMIC 的 TID，如果系统中 PMIC 总数少于 6 片，对应的寄存器需要写为 0。因此，各个 PMIC 之间的相互组合并不是随意的，TID 错误、重复、缺漏都会导致 SPMI 通信失败，应用中需要注意。

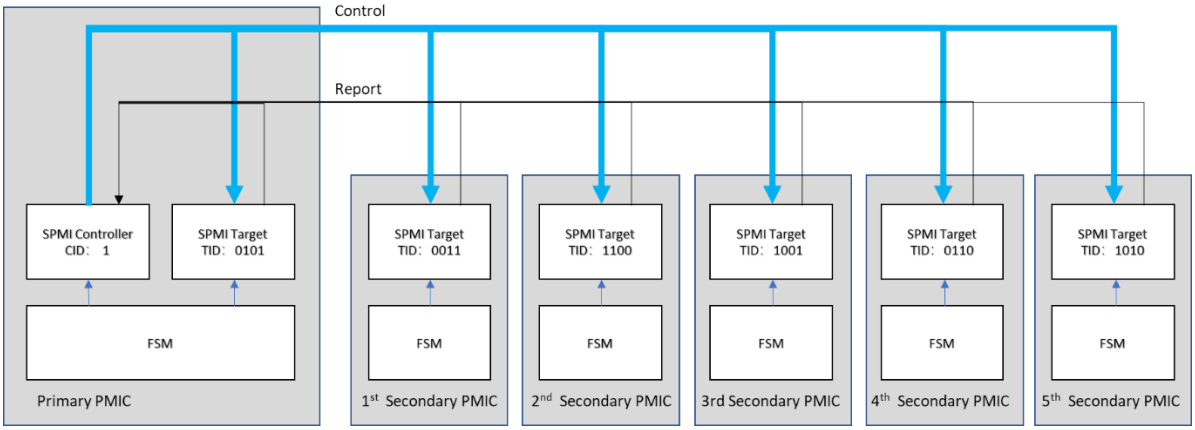

Figure 15. SPMI 框图

SPMI 总线上传递的信息有两种：

1. SPMI controller 和 SPMI target 之间相互同步各自 PFSM 的状态。
2. SPMI target 向 SPMI controller 传递 NVM ID。

SPMI target 仅在出现与 SPMI 无关的内部错误时与 SPMI controller 或其他 SPMI target 通信。当多个 target 发起通信时，根据 SPMI 规范中仲裁机制来确定优先级，仲裁获胜后，它将使用 GITD 1111 向所有 PMIC 传达 PFSM 的状态。

SPMI controller 会发起 SPMI BUS BIST 以检查 SPMI 通信的有效性和各个器件是否使用了正确的 NVM。SPMI controller 每隔一段时间（时长由寄存器 SPMI_WD_BOOT_INTERVAL 和 SPMI_WD_RUNTIME_INTERVAL 决定）发起一个广播命令，各 SPMI target 需要回复各自的 NVM ID，如果有任意一个 target 没有回复，controller 会不停尝试重发，直到 SPMI 模块内部看门狗超时（超时时长由 SPMI_WD_RESPONSE_TIMEOUT 寄存器设定，默认值在 820ms 左右），如图 16 所示是一个正常的 SPMI 通信波形，此 PDN 为 PDN 0A（TPS65941120+TPS65941421+LP876411B5），图中用橙色线标记了四段数据，第一段是 controller 发起的广播，后面三段数据是三个 target 回应的数据，四段数据缺一不可。可以通过逻辑分析仪或示波器测量 SPMI 信号的方式确认通信是否异常，一个常见的问题就是焊接故障导致 SPMI 通信异常，此时在波形上可以看到确实部分数据段。

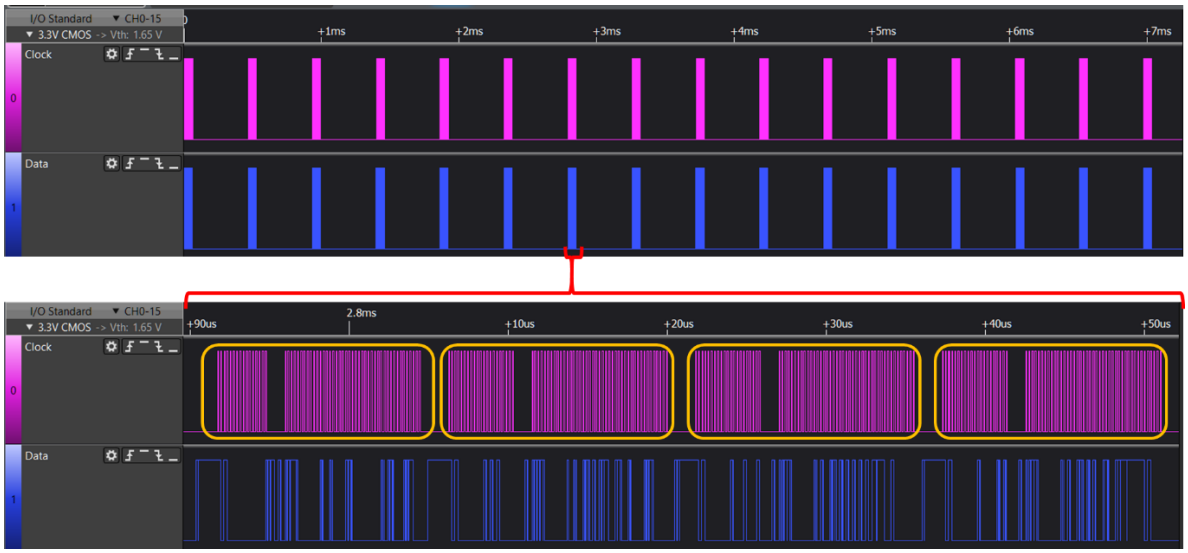

Figure 16. SPMI BIST 波形

3 参考文献

1. Datasheet “TPS6594-Q1 Power Management IC (PMIC) with 5 BUCKs and 4 LDOs for Safety-Relevant Automotive Applications”
2. Datasheet “LP8764-Q1 Four-Phase, 20-A Buck Converter With Integrated Switches”
3. Application Note “Scalable PMIC NVM Update Guide”
4. Application Note “TPS65941120-Q1, TPS65941421-Q1 and LP876411B5-Q1 PMIC User Guide for J721S2, PDN-0A”
5. Application Note “Optimized TPS65941213-Q1 and TPS65941111-Q1 PMI User Guide for Jacinto™ 7 J721E, PDN-0C”

重要通知和免责声明

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出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 产品的相关应用。 严禁以其他方式对这些资源进行复制或展示。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
版权所有 © 2025，德州仪器 (TI) 公司