

## STM32MP151/153/157 MPU 系列和 STPMIC1 集成在电池供电型应用中

### 简介

本应用笔记适用于 STM32MP151/153/157 MPU 系列器件，现在被称为 STM32MP15x。面向该应用的 STPMIC1 器件是 STPMIC1BPQR。

本应用笔记提供一个基于 STM32MP15x 和 STPMIC1BPQR 电源管理 IC 的硬件参考设计示例，由单体锂离子/锂聚合物电池供电。

本文档面向需要硬件集成和设置相关信息的产品架构师和设计人员，主要关注：

- 参考设计框图
- 功率分配
- 启动、关断和低功耗管理
- 电池管理（USB 充电和监测概述）
- USB 高速端口管理。

## 1 概述

本文档适用于 STM32MP15x 系列基于双核 Arm® 的微处理器。

注意

*Arm 是 Arm Limited (或其子公司) 在美国和/或其他地区的注册商标。*



## 2 概述

本应用笔记讲述 STM32MP15x 与 STPMIC1BPQR (带 DDR 和 Flash 存储器), 及以下外设一起使用:

- 直流输入电源来自单体锂离子电池 / 锂聚合物电池: 3.6 V 标称值 (2.8 V - 4.3 V)。
- 外设 I/O 接口电压为 1.8 V (由 STPMIC1BPQR 供电)。
- lpDDR2 或 lpDDR3 采用 32 位宽总线。
- DDR3/DDR3L 具有 x32 位总线宽度
- eMMC Flash 存储器由 2.9 V 电源供电, 并配备 1.8 V I/O 接口。
- USB 高速端口, 支持受电模式 (用于电池充电) 和供电模式 (为 USB 设备供电)。
- 以电池充电和电池监测为例。

该应用笔记不涵盖:

- **DDR3 和 DDR3L:** 由于这些存储设备的功耗比较高, 不适合与电池供电型产品一起使用。
- **I/O 电压为 3.3 V 的外设接口:** 假设电池电压降至 3.3 V 以下时, 单体锂离子电池 / 锂聚合物电池仍有足够的能量。如果使用 STPMIC1A 之类的外设, 它将无法在这些环境下工作。

### 2.1 参考文档

表 1. 参考文档

文档编号	标题
<b>STMicroelectronics 文档</b>	
[1]	STM32MP1 系列硬件开发入门 (AN5031)
[2]	高度集成的电源管理 IC, 用于微处理器单元 (DS12792)
[3]	STM32MP1 系列使用低功耗模式 (AN5109)
[4]	STM32MP151/153/157 系列与 STPMIC1A 硬件和软件集成 (AN5089)
[5]	基于 STM32MP157 高级 Arm® 的 32 位 MPU (RM0436)
[6]	Arm® 双 Cortex®-A7 800 MHz + Cortex®-M4 MPU、3D GPU、TFT/DSI、37 个通信接口、29 个定时器、广播模拟、密码 (DS12505)
[7]	独立式 USB Type-C™ 控制器带高压保护功能 (DS11503)
[8]	STM32MP1 系列寿命估计 (AN5438)
<b>USB 规范</b>	
[9]	USB Type-C 电缆与连接器规范版本 1.4 或更高版本可以从 USB 实现论坛网址获取

**注意** 如需意法半导体文档, 请参见 [www.st.com](http://www.st.com)。

### 3 词汇表

表 2. 词汇表

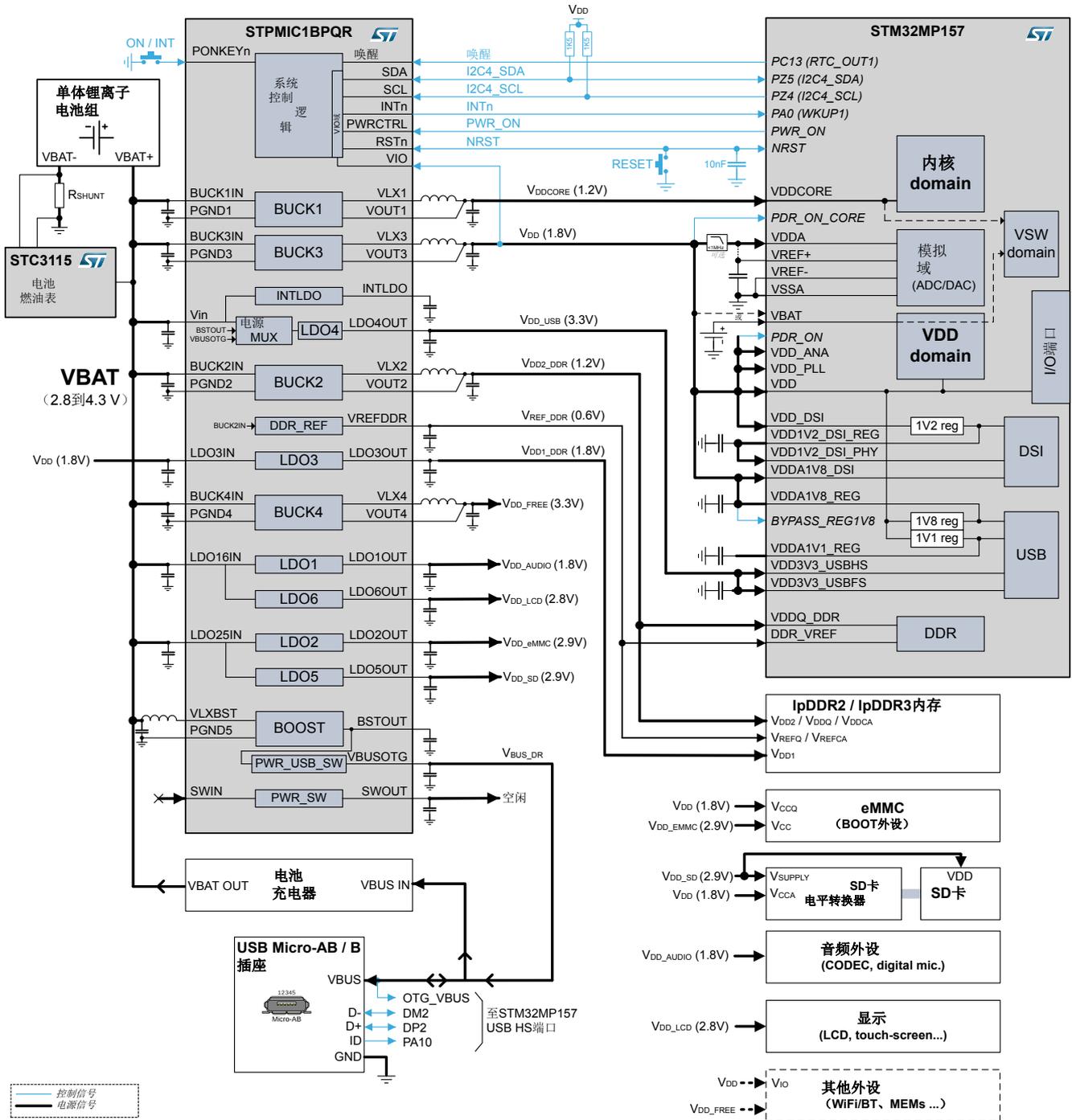
术语	定义
DR	双重功能。在本文档中，支持 USB 主机模式或 USB 设备模式的外设。
FSBL	第一阶段自举程序
HSI	高速内部振荡器
IC	集成电路
LDO	低压降线性稳压器
MPU	微处理器单元
PMIC	电源管理集成电路
RTC	实时时钟
SMPS	开关模式电源
SW	软件
TAMP	篡改检测电路

## 4 壁式适配器供电电池应用参考设计

该参考设计的目标应用由可重复充电的电池主电源适配器供电，配有低功耗 **lpDDR2** 或 **DDR3L**、一个 **eMMC**、一个 **SD-card**、一个 **USB HS** 双重功能数据端口（也称为 **OTG**）、一个启动 **flash**（**eMMC**、**NAND** 或 **NOR**），以及一个 **SD-card** 和两个 **USB 2.0 HS** 主机端口。为了详细介绍应用，还添加了其他外设（如音频设备、显示设备、无线器件、**MEMS** 等）。

**USB HS** 双重功能数据端口也用作电源，为电池充电。电池电量计监测电池的能量水平。为了详细介绍应用，还添加了音频设备、显示设备、**Wi-Fi**<sup>®</sup> / **蓝牙**<sup>®</sup> 传感器等其他外设。

主外设接口的 **I/O** 电压是 **1.8 V**。整个系统如图 1 中所示。

**图 1. STM32MP15x 和 STPMIC1BPQR (带 IpDDR2/3、eMMC、SD-card 和 USB DR)**


## 4.1 电池作为主电源

该应用由可充电的单体 3.6 V 标称电压锂离子或锂聚合物电池组 (VBAT) 供电，电压范围为 2.8 V - 4.3 V。图 1 中参考设计的电池容量取决于应用用例和外设功耗。例如，假设使用 500 mAh – 1000 mAh 电池组。

**警告** 在本文档中，STM32MP15x VBAT 引脚（用于 RTC/TAMP 供电）不能与为应用供电的电池电压域 VBAT 混淆。

电池组用作主电源 (VBAT)，为 STPMIC1BPQR 供电。STPMIC1BPQR 拥有所需的全部电源转换器，可为整个应用供电。

电池电压 (VBAT) 通过两个 STPMIC1BPQR 阈值进行监控：

- VINOK\_rise: 防止 STPMIC1BPQR 在电池电压 (VBAT) 过低时启动。  
STPMIC1BPQR VINOK\_rise 阈值 = 3.3V 标称电压。
- VINOK\_fall: 强制 STPMIC1BPQR 断电，防止电池完全放电（电量耗尽）。  
STPMIC1BPQR VINOK\_fall 阈值 = 2.8V 标称电压。  
VINOK\_rise 和 VINOK\_fall 阈值可以调整，方法是通过 I<sup>2</sup>C 对 STPMIC1BPQR NVM 重新编程。

STC3115 燃油表监测电池电压，并作为电流传感器，以准确估计剩余的电池电量水平。此外，如果电池组的位置靠近 STC3115，STC3115 还可以监测电池温度。

电池充电过程由一个分立式电池充电器 IC 管理。充电能量由 USB 连接器（Type Micro-AB 或 Micro-B）提供。充电器 IC 将 USB 电压 (VBUS) 转换为电池电压 (VBAT)。

## 4.2 功率分配

STPMIC1BPQR 集成了用于所有 STM32MP15x 供电的稳压器，以及一组为应用外设供电的稳压器。

### 4.2.1 VDD 电源域 (1.8 V)

VDD 是参考设计主 I/O 电压域，用于 STM32MP15x、STPMIC1BPQR 和外设。

它由 STPMIC1BPQR BUCK 3 降压 SMPS 供电，后者在所有负载条件下均具有高效率 and 低静态电流。

VDD 电压域是上电时序过程中第一个可用电压 (STPMIC1BPQR Rank1)，是下电时序过程中最后一个被禁用的。

VDD 在运行模式、LP-stop 模式、LPLV-stop 模式，以及待机模式下均被启用。VDD 在关闭模式下被禁用。

STPMIC1BPQR 在 STPMIC1BPQR 上电过程中，通过专用 NVM 设置将 BUCK 3 (I/O) 设为 1.8 V，以符合图 1 中的电池参考设计。详细信息，请参见[2]。

#### STM32MP15x VDD 电源：

除了为 STM32MP1 系列微处理器 VDD I/O 电压域供电，将 VDD\_PLL、VDD\_ANA、VDD\_DSI 和 VDDA1V8\_DSI 连接到 VDD。

ADC/DAC 模拟电压 (VDDA) 和相关模拟参考电压 (VREF+) 可以由 VDD 供电，具体取决于预期的 ADC 性能。如果期待较高的 ADC 性能或确保 DAC 正常工作，可使用电压高于 1.8 V 的低噪声电源为 VDDA（和 VREF+）供电。在这种情况下，STPMIC1BPQR LDO 可以专门用于为 STM32MP15x VDDA 电压域供电。

### 4.2.2 VDDCORE 电源域 (1.2 V / 1.35 V)

VDDCORE 是主 STM32MP15x 数字电源域。

它由 STPMIC1BPQR BUCK1 降压 SMPS 供电。该电压域是在上电时序中可用的第二个电源域 (STPMIC1BPQR Rank2)，也是断电过程中倒数第二个要被禁用的。

VDDCORE 在运行模式、LP-Stop 模式，以及 LPLV-Stop 模式下均被启用。VDDCORE 电压在 LPLV-stop 模式下会降低，以节约电源。VDDCORE 在待机和关闭模式下被禁用。

STPMIC1BPQR 通过 NVM 设置将 BUCK1 (VDDCORE) 配置为 1.2 V（当 STPMIC1BPQR 上电后）。

STM32MP15xD 和 STM32MP15xF 器件具有增强的用户任务概述（参见 [8]）。该配置文件允许 Arm® 双 Cortex®-A7 CPU 时钟频率最高可达 800 Mhz（参见 [6] 了解详细信息和限制）。

相应地，当 CPU 的工作频率 (Fmpuss\_ck) 达到 800 MHz 时，VDDCORE 电压必须提高到 1.35 V。当其不在 800 MHz 的运行模式下运行时，VDDCORE 电压必须设置回其标称电压（1.2 V 典型值）。

### 4.2.3 VDD\_USB 电源域 (3.3 V)

VDD\_USB 专门用于为 STM32MP15x USB PHY 供电 (VDD3V3\_USBHS 和 VDD3V3\_USBFS)。

它由 STPMIC1BPQR LDO4 线性稳压器供电，后者专为此功能而设计。

LDO4 配一个三路电源复用器，可以自动选择最高输入电压。其设计目的是让 STM32MP15xUSB PHY 在以下用例中工作：

- USB 内嵌主机具有低电池电压 ( $V_{BAT} \leq 3.3\text{ V}$ )：有 USB 外设（如大容量存储设备）连接到应用，由 STPMIC1BPQR 升压转换器通过 PWR\_USB\_SW 提供 VBUS\_DR (5.2 V) 电源（参见图 1）。在这种情况下，LDO4 不能从电池生成 3.3 V 的 VDD\_USB，因为 VBAT 低于 3.3 V。LDO4 输入电源多路复用器自动将 BSTOUT（BOOST 转换器提供 5.2 V 电压）设为 LDO4 输入电源，以生成 3.3 V 的 VDD\_USB。

**注意** 为了举例说明，忽略 LDO4 电压下降。

- USB 外设（设备）带低电池电压 ( $V_{BAT} \leq 3.3\text{ V}$ )：有 USB 主机（如来自 PC 的标准下行端口）连接到应用，提供 VBUS (5 V) 到 VBUS\_DR（参见图 1）。LDO4 不能从电池生成 3.3 V 的 VDD\_USB，因为 VBAT 低于 3.3 V。LDO4 输入电源多路复用器自动将 VBUSOTG（来自 VBUS\_DR）设为 LDO4 输入电源 (5 V)，以生成 3.3 V 的 VDD\_USB。

STPMIC1BPQR LDO4 多路复用器的电源可以是：VBAT（通过 VIN 引脚）、升压转换器（通过 BSTOUT 引脚），以及 VBUS\_DR（通过 VBUSOTG 引脚）。

该电压域是上电时序期间最后一个可用的域（STPMIC1BPQR Rank 3），也是下电时序期间第一个被禁用的域（除了被软件启用的稳压器是在 LDO4 之前被禁用，Rank 0。详细信息，请参见[2]）。

如果一个 USB 外设连接到应用，VDD\_USB 可以在运行模式、LP-stop 模式，或 LPLV-stop 模式下被启用。如果没有连接 USB 外设，它可以被禁用。VDD\_USB 在待机模式和关闭模式下被禁用。

STPMIC1BPQR 通过 NVM 将 LDO4 (VDD\_USB) 设为 3.3 V（当 STPMIC1BPQR 上电后）。上电时需要 LDO4 (VDD\_USB) 为 USB PHY 供电，用于 USB 刷写用例（STM32MP15x 外设从 ROM 启动）。

为了在上电时节约电源，对 STPMIC1BPQR NVM 重新编程，以便自动启用 LDO4（仅当 VBUS 在上电时存在于 VBUS\_DR 之上时）。

## VDD1\_DDR (1.8 V)、VDD2\_DDR (1.2 V)、VREF\_DDR (0.6 V) 电源域

### 4.2.4

- VDD1\_DDR 以 1.8 V 电压专门为 IpDDR2 或 IpDDR3 内核电源 1 (VDD1) 供电。
- VDD2\_DDR 以 1.2 V 电压专门为 IpDDR2 或 IpDDR3 内核电源 2 (VDD2 / VDDQ / VDDCA)，以及 STM32MP15x DDR I/O 电源 (VDDQ\_DDR) 供电。
- VREF\_DDR 以  $VDD2\_DDR / 2$  (0.6 V) 电压专门为 IpDDR2 或 IpDDR3 参考电压 (VREFQ / VREFCA) 和 STM32MP15x DDR 参考电压 (DDR\_VREF) 供电。
- VDD1\_DDR (1.8 V) 由 STPMIC1BPQR LDO3 供电。为了优化电源效率，LDO3 由 VDD (LDO3IN = VDD = 1.8 V) 供电。LDO3 被设计成也支持旁路模式（例如，电源开关），以较低的 RDSon 适应 IpDDR2 / IpDDR3 VDD1 电压公差。这同时考虑了 BUCK3 (VDD) 电压公差和 LDO3 RDSon 公差，以适应 IpDDR 的 VDD1。
- VDD2\_DDR (1.2 V) 由 STPMIC1BPQR BUCK2 降压 SMPS 供电，后者在所有负载条件下均具有高效率 and 低静态电流。BUCK2 由电池电压 (VBAT) 供电。
- VREF\_DDR (0.6 V) 由 STPMIC1BPQR REFDDR 受电/供电 LDO 供电。启用之后，REFDDR 输出电压等于  $VDD2\_DDR / 2$  (BUCK2 输出电压 / 2)。

STPMIC1BPQR 不会在上电时启动 VDD1\_DDR、VDD2\_DDR 和 VREF\_DDR。在 STM32MP15x 启动和关断时，它们必须由 STM32MP15x 软件控制上电和下电。

#### 软件 IpDDR2 / IpDDR3 上电时序：

上电之后，当 STM32MP15x 启动时，其软件必须按以下顺序启用 VDD1\_DDR、VDD2\_DDR 和 VREF\_DDR 电压域，以符合 IpDDR2 / IpDDR3 上电时序（参见 JESD209-2B 第 3.4 章节或最新 JEDEC 版本）：

1. 在旁路模式下启用 LDO3：VDD1\_DDR 在大约 100  $\mu\text{s}$  时间内上升到 1.8 V (= VDD 电压)。
2. 100  $\mu\text{s}$  之后，启用 REFDDR LDO：VREF\_DDR 电压保持在 0 V (BUCK2 被禁用)。
3. 将 BUCK2 设为 1.2 V，且启用 BUCK2。

限制条件是步骤 1 到步骤 3 必须在 20 ms 之内执行，以符合 IpDDR2 / IpDDR3 JEDEC 约束。

### 软件 IpDDR2 / IpDDR3 下电时序:

在 STM32MP15x 软件进入关闭模式或待机模式（假设软件策略是在待机模式下关闭 DDR）之前，STM32MP15x 软件必须以下列顺序管理 IpDDR2 / IpDDR3 下电:

1. 软件接收事件，然后进入关闭或待机模式。
2. 将 IpDDR CKE 置为低电平。
3. 禁用 BUCK2:
  - a. VDD2\_DDR 在 1.5 ms 之内下降（由于 BUCK2 慢速下拉放电电阻）。
  - b. VREF\_DDR 电压紧随 VDD2\_DDR / 2（REFDDR 是推-挽 LDO）。
4. 等待 1.5 ms，禁用 REFDDR（VREF\_DDR 已经是 0 V，因为电压在之前的步骤已经下降）。
5. 禁用 LDO3: VDD1\_DDR 在 3 ms 之内下降（得益于 LDO3 的下拉放电电阻）。  
限制条件是步骤 3 到步骤 5 必须在 20 ms 之内执行，以符合 IpDDR2 / IpDDR3 JEDEC 约束。

### IpDDR2 / IpDDR3 非受控断电时序:

非受控断电时序通常在电池电压移除或复位时出现。在这样的情况下，STPMIC1BPQR 管理断电时序:

1. 电池被移除，VBAT 下降。
  2. VBAT 超过 VINOK\_fall: STPMIC1BPQR 关闭条件。
  3. STPMIC1BPQR 启动断电时序:
    - a. STPMIC1BPQR 确认复位 (NRST) → STM32MP15x 确认 DDR\_CKE 引脚未低电平且 DDR\_CLKP / DDR\_CLKN 时钟信号已停止。  
只要 STPMIC1BPQR 确认 NRST，复位信号传播到 STM32MP15x: DDR\_CKE 被设为低电平，DDR\_CKP 和 DDR\_CKN 已停止: IpDDR IC 功耗立刻下降。
    - b. STPMIC1BPQR 禁用 Rank0 稳压器: BUCK2、LDO3 和 REFDDR:
      - i. 在 BUCK2、LDO3、REFDDR 上设置下拉放电电阻。
      - ii. VDD1\_DDR、VDD2\_DDR、VREF\_DDR 电压下降。VDD2\_DDR 下降速度更快，因为下拉放电电阻大于 VDD1\_DDR 和 VREF\_DDR。  
BUCK2、LDO3 和 REFDDR 在复位之后由 STPMIC1BPQR 立刻禁用 (Rank0)，因为这些稳压器是在上电之后最后（通过软件）启用的。详细信息，请参见[2]。
  4. STPMIC1BPQR 先禁用 Rank3，然后是 Rank2，最后是 Rank1 稳压器（参见[2]获取详细信息）
  5. STPMIC1BPQR 关闭（或者无电源）。
- 参见第 5.2.3 节获取详细信息和限制条件。

#### 4.2.5 VDD\_eMMC 电源域 (2.9 V)

VDD\_eMMC 专门为 eMMC Flash 存储器内核域供电 (VCC)。eMMC Flash 存储设备也必须由 VDD 电压供电，以变为其 I/O 电源域 (VCCQ) 供电。

CLK、CMD 和 D0 必须在 VDD 上放置上拉电阻，以避免当 VDD\_eMMC 关闭时 VDDQ 上出现额外的 eMMC 功耗。

VDD\_eMMC 由 STPMIC1BPQR LDO2 线性稳压器供电。该电压域是在上电时序中第二个要启用的域 (STPMIC1BPQR Rank2)，也是断电过程中倒数第二个要被禁用的域。

如果 STM32MP15x 软件需要读写访问，VDD\_eMMC 在运行、LP-stop 和 LPLV-stop 模式下被启用。如果不期望进行读/写访问，软件必须禁用 VDD\_eMMC。VDD\_eMMC 在待机模式和关闭模式下被禁用。

STPMIC1BPQR 通过 NVM 将 LDO2 (VDD\_eMMC) 设为 2.9 V (当 STPMIC1BPQR 上电后)。在上电时需要 LDO2 (VDD\_eMMC) 为 eMMC Flash 设备供电，允许 STM32MP15x 从 ROM 启动时访问该内存。

VDD\_eMMC 可以由应用软件在运行时打开或关闭。

如果 eMMC 设备是启动 Flash 外设，在应用进入待机模式之前，应用软件必须对 STPMIC1BPQR 进行编程，以便在待机模式 (PWR\_ON 信号低电平) 关闭 eMMC，在运行模式 (PWR\_ON 信号高电平) 对 eMMC 通电。在这种情况下，当应用程序从待机模式恢复到运行模式时，eMMC 上电并准备接受 STM32MP15x 启动 ROM (外设启动) 的访问。

#### 4.2.6 VDD\_SD 电源域 (2.9 V)

VDD\_SD 专门为 SD-card 设备和 SD-card 电平转换器 (VSUPPLY) 供电。SD-card 电平转换器设备也可以从 VDD 电压获得电源，为其 I/O 电源域 (VCCA) 供电。

它由 STPMIC1B LDO5 线性稳压器供电。

该电压域是在上电时序中第二个要启用的域 (STPMIC1BPQR Rank2)，也是断电过程中倒数第二个要被禁用的域。

VDD\_SD 由软件管理，可以在运行时开启或关闭。

STPMIC1B 通过 NVM 设置将 LDO5 (VDD\_SD) 设为 2.9 V (当 STPMIC1B 上电后)。在上电时需要 LDO5 (VDD\_SD) 为 SD-card 及其电平转换器 IC 供电，允许 STM32MP15x 从 ROM 启动时访问该内存。

VDD\_SD 可以由应用软件在运行时打开或关闭。

如果 SD-card 设备是启动 Flash 外设，在应用进入待机模式之前，应用软件必须对 STPMIC1B 进行编程，以便在待机模式 (PWR\_ON 信号低电平) 下关闭 SD-card，在运行模式 (PWR\_ON 信号高电平) 对 SD-card 通电。在这种情况下，当应用程序从待机模式恢复到运行模式时，SD-card 上电并准备接受 STM32MP15x 启动 ROM (外设启动) 的访问。

#### 4.2.7 VBUS\_DR 电源域 (5 V)

VBUS\_DR 是 USB 高速接口的专用电源域。VBUS\_DR 连接到 USB 插座的 VBUS 引脚，并具有双向电流：

- 应用处于 USB 设备模式：USB 主机外设连接到应用，并为应用提供 VBUS\_DR 电压 (参见图 2)。
- 应用处于 USB 主机模式：USB 设备外设连接到应用，并从应用获得 VBUS\_DR 电压 (参见图 3)。

图 2. USB 设备模式下的 VBUS\_DR 电源路径

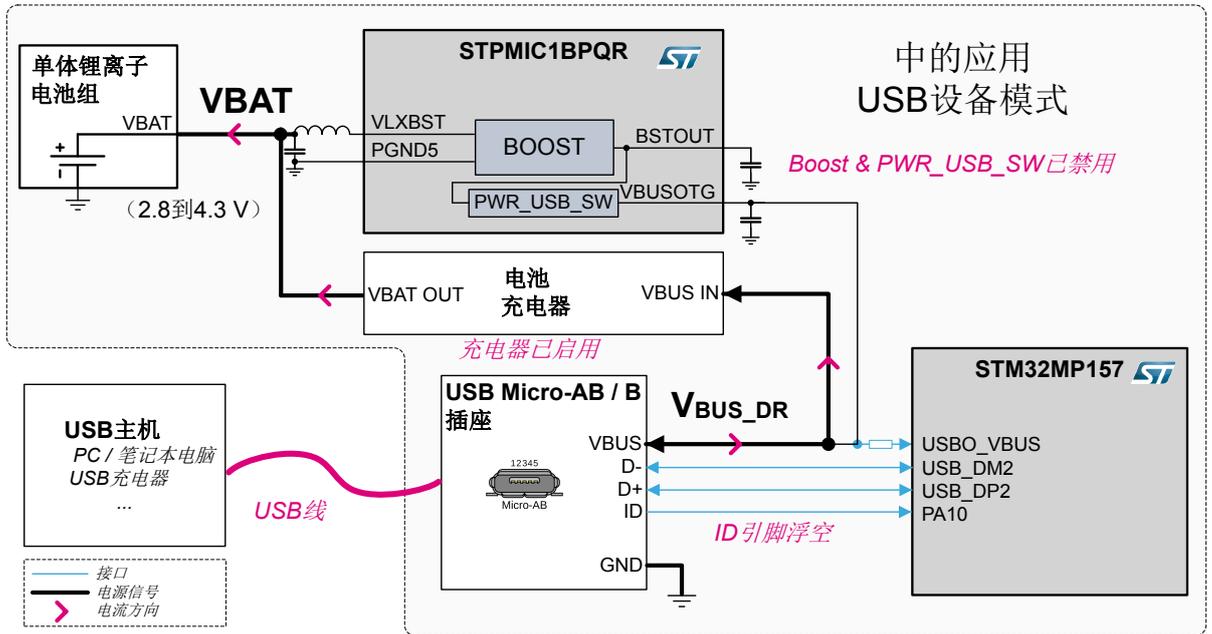
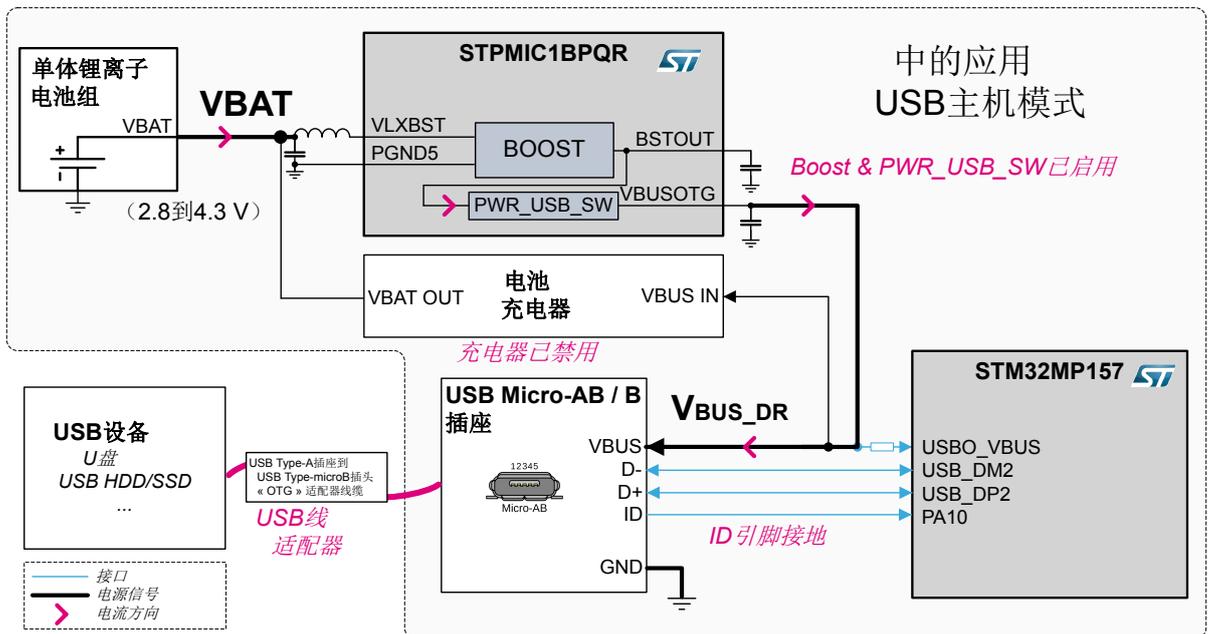


图 3. USB 主机模式下的 VBUS\_DRD 电源路径



#### 4.2.8 VDD\_AUDIO (1.8 V)、VDD\_LCD (2.8 V) 电源域

提供 VDD\_AUDIO 和 VDD\_LCD，用于演示图 1 中的参考设计。它们分别由 LDO1 和 LDO6 线性稳压器供电。在上电时，VDD\_AUDIO 和 VDD\_LCD 不由 STPMIC1BPQR 启用。如果相关外设需要它们，则在上电后由软件启用并设为正确的电压值。

VDD\_AUDIO 和 VDD\_LCD 由软件管理，可以在运行时开启或关闭。在待机模式和关闭模式下，VDD\_AUDIO 和 VDD\_LCD 被禁用。

### 4.3 控制信号与接口，用于连接 STM32MP157 和 STPMIC1BPQR

本节概述 STM32MP157 微处理器与 STPMIC1BPQR 设备的通信方式。有多个接口可供选择，具体取决于应用需求。本文档的第一部分对每个接口进行了描述。

#### I<sup>2</sup>C 接口：

STPMIC1BPQR 可以由 STM32MP15x 通过 I<sup>2</sup>C 接口进行控制，以便：

- 启用或禁用调节器
- 设置调节器电压和模式（低功耗或高功率）
- 设置低功耗管理（PWRCTRL 行为）
- 设置中断控制器或读取中断状态
- 设置保护（看门狗、过电流、欠压）或读取保护状态。
- 对 NVM 重新编程，以更改启动行为。

STPMIC1BPQR 具有特殊的默认 NVM 设置，允许来自 USB 接口（用于刷写或加载，然后执行软件）、SD-card 或 Flash 存储器（比如 eMMC）的 1.8 V I/O 启动 STM32MP15x 应用程序。一旦 STM32MP15x 能够执行软件，它也能够对 STPMIC1BPQR NVM 进行动态重新编程，以便对应用程序进行微调。

#### ON / INT 按钮：

“ON / INT”用户按钮连接到 STPMIC1BPQR PONKEYn 引脚（低电平有效）。该按钮允许：

- 上电 STPMIC1BPQR。
- 当应用程序运行时，就按钮按下事件或按钮释放事件发送中断给 STM32MP15x。
- 长按（默认为 16s）以强制关闭 STPMIC1BPQR。

#### NRST 信号：

NRST 是双向低电平有效的信号，用于 STM32MP15x 和 STPMIC1BPQR。STM32MP15x NRST 引脚和 STPMIC1BPQR RSTn 引脚采用数字输入 / 开-漏输出拓扑：

- 当 STPMIC1BPQR 将 RSTn 置为有效后（例如在上电或下电时序期间），它将 NRST 信号驱动为低电平；STM32MP15x 被强制进入复位状态，直至 STPMIC1BPQR 释放 NRST。
- 当 STM32MP15x 将 NRST 信号（例如 STM32MP15x 看门狗复位）置为有效，或用户按下“复位”按钮，STPMIC1BPQR 立即将 RSTn 引脚置为有效并执行不可中断的电源循环：STPMIC1BPQR 执行下电时序，然后是上电时序，最后释放 RSTn。  
在断电重启时序的最后，STPMIC1BPQR 等待 NRST 信号在重新复位之前变为高电平，避免无限的复位循环。

#### INTn 信号：

INTn 是一个 STPMIC1BPQR 输出低电平有效中断线，连接到 STM32MP15x PA0 输入引脚。PA0 兼具中断和唤醒能力：

- 当 STM32MP15x 处于运行模式或停止模式时，管理来自 STPMIC1BPQR 的中断。
- 当 STM32MP15x 处于待机模式时，将其唤醒。

#### PWR\_ON 信号：

PWR\_ON 信号由 STM32MP15x PWR\_ON 引脚驱动，用于控制 STPMIC1BPQR PWRCTRL 引脚。

如此一来，STM32MP15x 便能非常快速地将 STPMIC1BPQR 电源策略切换到以下应用程序功耗模式之一：

- 从运行模式切换到 LPLV\_Stop 模式，以及切换回原有模式
- 从运行模式切换到待机模式，以及切换回原有模式。

（参见[3]详细了解使用 STPMIC1BPQR 时的低功耗模式管理和 PWR\_ON 引脚设置）

上电或复位之后，STPMIC1BPQR PWRCTRL 引脚被禁用。在进入低功耗模式之前，STM32MP15x 通过 I<sup>2</sup>C 设置 STPMIC1BPQR，根据 PWR\_ON 信号状态对预期的电源行为进行编程。

#### EADLY 定时器

EADLY 定时器防止启动 ROM 在准备就绪之前对启动外设执行任何访问。等待 Flash 存储器（eMMC 或 SD 卡）上的电压稳定下来，确保启动软件被启动 ROM 可靠地读取。复位之后的默认延迟时间是 10 ms（参见[5]获取详细信息）。

## POPL 定时器

STM32MP15x POPL 定时器允许 STM32MP15x 保持在待机状态，并在最短持续时间内将 PWR\_ON 信号置为低电平，允许外设稳压器在重新启动之前停止。这是为了确保在应用程序进入待机状态后发生唤醒事件时，外设能够正确重启。STPMIC1BPQR 为每个稳压器输出配一个放电电阻，允许所有稳压器输出电压在 3 ms 时间内放电。所以，POPL 可以为 3 ms 最小值，或者保持默认值 (10 ms)，前提是不必重点关注待机状态唤醒的持续时间。

### 唤醒信号 (可选)：

唤醒信号由 STM32MP15x PC13 (RTC\_OUT) 引脚驱动，用于控制 STPMIC1BPQR 唤醒引脚。它允许 STM32MP15x 为 STPMIC1BPQR 上电；通常，在实时时钟定时器结束时。

如果纽扣电池连接到 STM32MP15xVBAT 引脚，则该功能可用。

### 4.3.1 采用可移除电池组的应用中的唤醒引脚管理

如果纽扣电池连接到 STM32MP1 系列微处理器 VBAT 引脚，而产品由可移动电池组供电，则必须实现图 4 中描述的电路图。

图 4. 可移除电池组应用的唤醒引脚控制电路

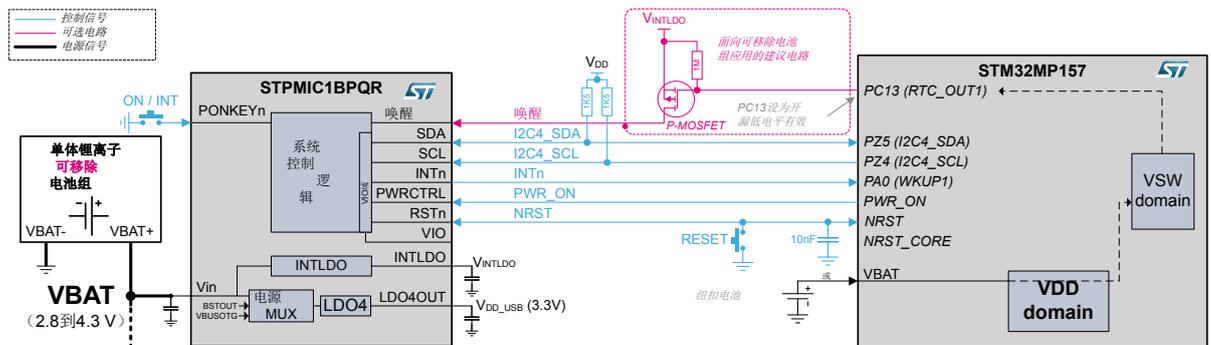


图 4 中的电路将 STM32MP1 系列微处理器 PC13 IO 电压与 STPMIC1BPQR 唤醒引脚分隔。这样可以防止 STM32MP1 系列微处理器 PC13 IO 在以下情况发生电流泄漏：

- 通过 STPMIC1BPQR 唤醒引脚将 PC13 设为 STPMIC1BPQR VIN。
- 电池组被移除。
- VBAT 电压低于 PC13 电压。

当使用该电路实现时 (参见图 4)，STM32MP1 系列微处理器 PC13 必须设置为开漏低电平有效 I/O。必须选择 P-MOSFET，以实现低于 1.8 V (低于 VINTLDO) 的最小 Vgs 阈值。P-MOSFET 能够维持大于 40  $\mu$ A 的漏极电流 (ID) 来驱动唤醒引脚。

注意

STPMIC1BPQR 唤醒引脚拥有内部下拉电阻器 ( $45\text{K}\Omega < R_{pd} < 80\text{K}\Omega$ )，内部连接到 GND。

## 5 电源管理

回顾以下功耗模式：

- 工作模式
- 应用上电和下电模式
- 低功耗管理模式
- 用户复位和崩溃恢复管理
- 软件管理示例

### 5.1 工作模式

应用程序可以根据系统活动切换到不同的工作模式。工作模式由 STM32MP15x 管理，它们控制电源管理和时钟分配（参见[3]获取详细信息）。

表 3 总结应用级别的工作模式。STPMIC1BPQR 功耗模式取决于应用程序工作模式。

表 3. 应用程序工作模式

工作模式	STPMIC1BPQR 功耗模式	VBAT <sup>(1)</sup>	PWR_ON	说明	注释
运行	Power-on-main	> VINOK_fall	1	VDD 上电 VDDCORE 上电，系统时钟开启 lpDDR 主动/自动刷新 外设上电 / 下电	(2)
停机	Power-on-main	> VINOK_fall	1	VDD 上电 VDDCORE 上电，系统时钟关闭 lpDDR 自刷新 外设上电 / 下电	(2)
LPLV-stop	Power-on-alternate	> VINOK_fall	0	VDD 上电 VDDCORE 以低电压上电，系统时钟关闭 lpDDR 自刷新 外设上电 / 下电	(3)
待机	Power-on-alternate	> VINOK_fall	0	VDD 上电 VDDCORE 下电，系统时钟关闭 lpDDR 自刷新/关闭 外设关闭	(3)
断电	关闭	> VIN_POR_fall	-	全部断电	-
	No_supply	< VIN_POR_fall	-	全部断电	-
Coin-cell-VBAT	No_supply	< VIN_POR_fall	-	全部断电（除了 STM32MP1 系列微处理器 VSW）	(4)

1. STPMIC1BPQR 硬件阈值。详细信息，请参见[2]。
2. 运行模式和停止模式的区别仅基于 STM32MP1 系列微处理器时钟管理。对于电源管理，运行模式和停止模式没有区别。
3. 从运行模式进入 LPLV\_Stop 模式或待机模式时，PWR\_ON 控制引脚没有区别（两种情况下，PWR\_ON 信号均从高电平到低电平）。进入 LPLV\_Stop 模式或待机模式之前，STM32MP1 系列微处理器通过 I<sup>2</sup>C 接口对 STPMIC1BPQR 进行编程，对稳压器进行相应设置。
4. 为了在 VDD 关闭后保留 STM32MP1 系列微处理器 VSW 域（RTC、备份寄存器、备份 RAM 和保留 RAM）的内容，可以将 STM32MP1 系列微处理器 VBAT 引脚连接到可选纽扣电池。

#### 5.1.1 应用通电 / 断电条件

当应用处于断电模式时，需要一个通电条件使 STPMIC1BPQR 上电并进入运行模式。类似地，如果应用需要进入断电模式，需要一个关闭条件才能使 STPMIC1BPQR 下电。

当分别出现打开或关闭条件时，STPMIC1BPQR 自主管理上电和下电时序（参见[2]获取详细信息）。

### 打开条件

当供电电源 (VIN) 上升并高于 VINOK\_rise 时，STPMIC1BPQR 自动上电（在 STPMIC1BPQRNVM 中默认启用的自动打开特性）。如果 STPMIC1BPQR 处于关闭状态（且 VIN > VINOK\_rise），可以通过三种外部触发条件之一获得供电：

- “ON / INT”用户按钮被按下：PONKEYn 引脚电压下降沿。
- USB 主机或 USB 充电器线缆插入：VBUSOTG 引脚电压上升沿。
- 发生 STM32MP15x 唤醒事件（例如，通过 STM32MP15x Pc13 引脚实现 RTC 或篡改唤醒）：唤醒引脚电压上升沿。

**注意** 如果纽扣电池连接到 STM32MP15xVBAT 引脚，则 STM32MP157 唤醒功能可用。

上电后，STPMIC1BPQR 经历暂时上电状态，其中稳压器以预定义的顺序（等级）和电压依次启动，并通过释放 NRST 信号来表示结束。在此状态之后，STPMIC1BPQR 进入通电状态并保持，应用程序现在可以运行。该状态可以从关闭模式（当通电条件出现时）或 NO\_SUPPLY 模式（当 VBAT 电压上升并高于 VINOK\_rise，触发‘自动通电’功能时）达到。

### Turn-OFF 条件

关闭条件的出现使 STPMIC1BPQR 下电并进入关闭状态。在关闭状态下，所有稳压器均关闭。出现以下情形时，通电状态下的 STPMIC1BPQR 将被断电：

- 软件关闭：STM32MP15x 发送 I2C 命令到 STPMIC1BPQR。
- 长按“ON/INT”用户按钮：按下复位按钮并保持 16 秒，STPMIC1BPQR 将关闭（延迟可编程）。
- 热关断：如果发生过热，则 STPMIC1BPQR 关断，并会在温度回到正常水平后重启。
- 过流保护：如果通过软件启用稳压器，则相应产生的过流条件会导致 STPMIC1BPQR 关断。
- 看门狗：如果是通过软件启用的，则当倒计时器到达 0 时，STPMIC1BPQR 进入关闭状态。
- VINOK\_fall：如果 VBAT 降至 VINOK\_fall 阈值以下，则 STPMIC1BPQR 进入关闭状态。

下电时，STPMIC1BPQR 经历暂时下电状态，NRST 被置为有效，导致稳压器以与上电顺序相反的顺序停止。在此状态之后，STPMIC1BPQR 处于关闭状态并保持这种状态，直至通电条件出现。该状态是从通电状态下通过断电条件的出现而实现的。

需要指出的是，只要某个 STPMIC1BPQR 稳压器被禁用，则将在其输出上启用一个下拉电阻，以释放解耦电容器电压。LDO 和 BUCK 稳压器输出电压分别在 3 ms 和 1.5 ms 时间内放电（参见[2]获取详细信息）。

应用程序可以将 STPMIC1BPQR“重新启动请求”功能设置为：在关闭条件出现后自动重新启动应用程序（请参阅[2]获取详细信息）。

## 5.1.2 STPMIC1BPQR restart\_request 和 mask\_reset 选项

在关闭条件出现之前，STM32MP15x 软件可以通过编程让 STPMIC1BPQR 重启，而不是将其关闭（在 STPMIC1BPQR 中设置 restart\_request 功能）。此设置必须在关闭条件出现之前完成；例如在应用程序上电之后。

例如，软件可在 STPMIC1BPQR 中设置重启请求位 (RREQ\_EN = 1)，以彻底重新启动应用程序，然后通过编程实现软件开关关闭 (SWOFF = 1)。STPMIC1BPQR 将执行断电重启时序：首先是断电时序（禁用所有稳压器），然后是上电时序（重启稳压器，再释放 NRST 信号）。

如果应用程序需要在断电重启期间让一个或多个 STPMIC1BPQR 稳压器保持启用状态，则 STM32MP15x 软件可以对 STPMIC1BPQRmask\_reset 选项进行编程（方法是设置面向降压转换器的 STPMIC1BPQRBUCKS\_MRST\_CR 寄存器，以及面向 LDO 的 LDOS\_MRST\_CR 寄存器）（参见[2]了解关于 STPMIC1BPQR mask\_reset 选项的详细信息）。此设置必须在断电重启之前完成；例如在应用程序上电之后。

为 STM32MP15xVDD 电源域供电的 BUCK3 就是典型情形。VDD 上的断电重启必须被屏蔽 (BUCKS\_MRST\_CR[2] = 1)，以防止丢失：

- STM32MP15x 备份 RAM
- 保留 RAM
- 备份寄存器内容。

如未设置 BUCKS\_MRST\_CR[2]，则当来自 STM32MP15x（参见第 5.4 节）的 NRST 或关闭条件（restart\_request 位已启用）触发断电重启进程时，该信息将丢失。

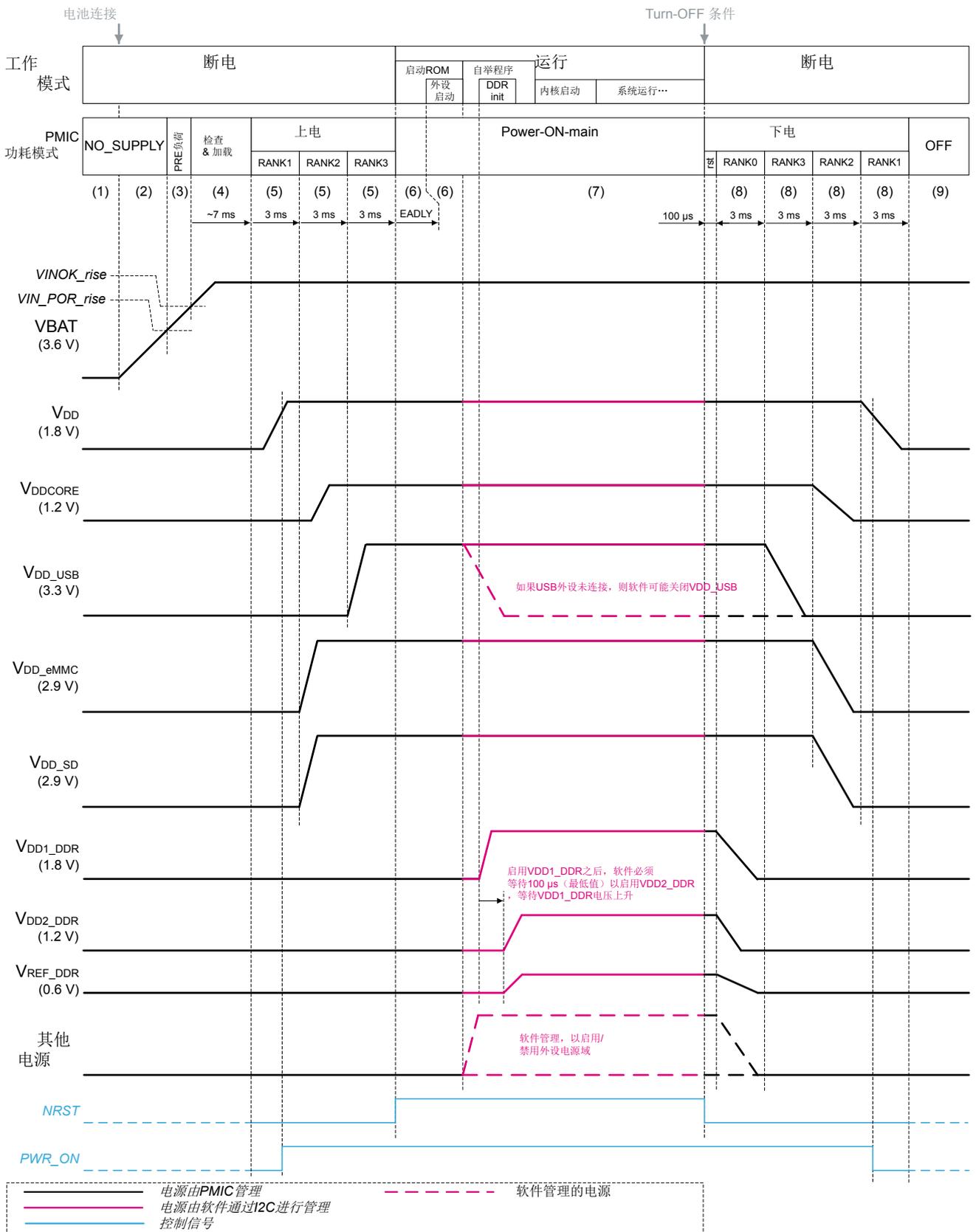
## 5.2 应用上电 / 下电时序

上电时序是由 STPMIC1BPQR 管理的位于断电模式和运行工作模式之间的暂时状态，下电时序也是如此。图 5 中显示应用上电和下电时序，详情请参见图 1 中的参考设计。

### 5.2.1 通过电池插入实现上电

应用处于关闭状态，未连接任何电源。连接电池，在 VBAT 上升后，应用自动启动（STPMIC1BPQR 默认在其 NVM 中启用‘自动通电’功能）。在 STPMIC1BPQR 上电后，应用启动（包括 lpDDR 初始化），最终系统达到运行模式。断电条件出现时，STPMIC1BPQR 下电并进入关闭模式，应用程序进入断电模式。整个过程如下并在图 5 中详细说明：

1. 应用无电源，或者 STM32MP15x 处于 coin-cell-VBAT 模式（从纽扣电池获取电源，为 STM32MP15x VSW 供电）。
2. 将充满电的电池组 (VBAT > VINOK\_rise) 连接到应用。VBAT 电压上升。
3. 一旦 VBAT 电压高于 VIN\_POR\_rise (STPMIC1BPQR VIN\_POR\_rise 阈值最初设为 2.3 V)：
  - a. STPMIC1BPQR 初始化并预加载其 NVM 内容。
  - b. STPMIC1BPQR 将 NRST 置为有效。
4. VBAT 电压上升并高于 VINOK\_rise，STPMIC1BPQR 检查通电条件（‘自动通电’功能已在 STPMIC1BPQR NVM 中启用）。一旦检测到有效的通电条件，STPMIC1BPQR 启动上电时序。
5. STPMIC1BPQR 紧随上电时序：
  - a. Rank1: BUCK3 (VDD) 被启用为 1.8 V，并等待 3 ms。
  - b. Rank2: BUCK1 (VDDCORE) 启用为 1.2 V。LDO2 (VDD\_eMMC) 和 LDO5 (VDD\_SD) 启用为 2.9 V 并等待 3 ms。STM32MP15x 现在执行内部初始化并释放其复位，STM32MP15x 保持复位状态（只要 STPMIC1BPQR 仍然确认 NRST 信号有效）。
  - c. Rank3: LDO4 (VDD\_USB) 被启用为 3.3 V（硬设置）。3 ms 过后，STPMIC1BPQR 释放 NRST 信号。
6. 随着 NRST 信号上升，STM32MP15x 和 STPMIC1BPQR 释放各自的复位引脚：
  - a. STM32MP15xEADLY 延时定时器 (10 ms) 将启动。更多信息，请参见 EADLY 定时器节。
  - b. 当 EADLY 延时过后，启动 ROM 开始访问外部外设（eMMC 或 SD-card，具体取决于 STM32MP15x 启动引脚设置），以加载和执行引导软件。
  - c. 自举程序可以控制任意 STPMIC1BPQR 稳压器（比如初始化 LCD 和绘制启动画面）。
7. 自举程序初始化 DDR，然后加载并执行内核：
  - a. LDO3 (VDD1\_DDR) 在旁路模式下被启用。软件至少再等待 100  $\mu$ s。
  - b. DDR\_VREF (VREF\_DDR) 被启用。
  - c. BUCK2 (VDD2\_DDR) 被启用为 1.2 V。软件等待至少 1.4 ms，然后 BUCK2 就绪。
  - d. 软件初始化 STM32MP15xDDR 控制器和 lpDDR 器件。
  - e. 自举程序加载并执行内核。内核初始化。
  - f. 系统进入运行状态。
8. 关闭条件出现。STPMIC1BPQR 执行下电时序：
  - a. STPMIC1BPQR 将 NRST 置为有效 (STM32MP15x 复位) 并等待 100  $\mu$ s。
  - b. Rank0: STPMIC1BPQR 禁用所有在上电时未启用的稳压器 (LDO3、BUCK2、DDR\_REF、LDO1、LDO6、BOOST、PWR\_USB\_SW、PWR\_SW) 并等待 3 ms。同样请参见 Turn-OFF 条件章节。
  - c. Rank3: LDO4 (VDD\_USB) 并禁用并等待 3 ms。
  - d. Rank2: BUCK1 (VDDCORE)、LDO2 (VDD\_eMMC) 和 LDO5 (VDD\_SD) 被禁用并等待 3 ms。
  - e. Rank1: BUCK3 (VDD) 被禁用并等待 3 ms。
9. STPMIC1BPQR 现在处于关闭模式：应用处于断电状态。

**图 5. 上电/下电序列**


### 5.2.2 从 STPMIC1BPQR 关闭模式上电

图 5 中的应用从 NO\_SUPPLY 状态开始上电，此时的通电条件是电池插入（自动通电功能在 STPMIC1BPQR NVM 中启用）。

从 STPMIC1BPQR 关闭模式上电的顺序与图 5 中从关闭模式上电的顺序类似。不同之处在于，当通电条件出现时，顺序从步骤(4)“检查 & 加载”开始，而不是等待 VBAT 电压上升。

区别如下所示：

- STPMIC1BPQR 最初由 VBAT 电压（高于 VINOK\_rise）供电，允许 STPMIC1BPQR 上电（而不是通过电池插入触发图 5 中的 VBAT 上升）。
- 图 5 的步骤(1)、(2)、(3) 被单一步骤取代；将 STPMIC1BPQR 功耗模式的“NO-SUPPLY”和“PRE-LOAD”合并为“OFF”。
- “电池连接”事件被“通电条件”事件取代，并置于 STPMIC1BPQR 状态的“OFF”和“CHECK & LOAD”之间。

### 5.2.3 电池移除导致下电

图 5 中的应用在关闭条件出现后断电，而 VBAT 保持有效电压。

如果电池移除导致应用断电，关闭条件是 VBAT 电压降至 VINOK\_fall 以下。一旦 VBAT 电源低于 VINOK\_fall，STPMIC1BPQR 在 100 $\mu$ s 内将 NRST 置为有效，然后按图 5 中步骤 (8) 所示关闭电源。

限制：电池移除之后，VBAT 在几毫秒之内非常迅速地降至 VINOK\_fall 值（取决于系统活动），只有这样，下电时序才开始。一旦 STPMIC1BPQR 将 NRST 置为有效，系统活动立即停止，功耗降低，VBAT 的下降也放缓。然而，在下电时序结束之前，VBAT 可能会降至 VIN\_POR\_fall 阈值以下。在这种情况下，STPMIC1BPQR 稳压器下拉放电电阻不再由 STPMIC1BPQR 控制。可以在 VBAT 路径上插大型解耦电容器（几百  $\mu$ F）以限制 VBAT 下降速度。

该时序很重要，主要用于管理“lpDDR2 / lpDDR3 不可控的断电时序”（详情请见第 4.2.4 节）。

## 5.3 低功耗 模式管理

STM32MP15x 支持多种工作模式，以降低功耗（参见第 5.1 节）。本节介绍 LPLV-stop 和待机这两种低功耗模式（参见[3]获取详细信息）。

**注意**

停止模式只涉及 STM32MP15xMPU 内部时钟管理，不涉及外部电源管理。所以，本节不介绍停止模式。

低功耗模式由 STM32MP15x 管理。STM32MP15x PWR\_ON 输出引脚连接到 STPMIC1BPQR PWRCTRL 输入引脚。然后可以切换 STPMIC1BPQR 状态：从 Power-ON-main 切换到 Power-ON-alternate，反之亦然。

上电之后，STPMIC1BPQR 进入 Power-ON-main 状态，直至 STM32MP15x 告知 STPMIC1BPQR 将 PWRCTRL 引脚激活为低电平有效（方法是在 STPMIC1BPQR MAIN\_CR 寄存器中设置 PWRCTRL\_POL = 0 且 PWRCTRL = 1）。

表 3 根据 PWR\_ON 信号状态，总结 STPMIC1BPQR 状态和相应的工作模式。

当 STPMIC1BPQR 从 Power-ON-main 转为 Power-ON-alternate，它从内部将 MAIN 控制寄存器 (xxxx\_MAIN\_CR) 的内容切换为 ALTERNATE 控制寄存器 (xxxx\_ALT\_CR) 的内容，反之亦然（参见[2]获取详细信息）。

在进入低功耗模式之前，STM32MP15x 必须根据预期的 STPMIC1BPQR 稳压器设置（针对低功耗模式行为）设置 STPMIC1BPQR ALTERNATE 控制寄存器。如果需要，STM32MP15x 必须设置 STPMIC1BPQR MAIN control 寄存器，确保应用退出低功耗模式。

### 5.3.1 LPLV-stop 模式

图 6 中显示应用 LPLV-stop 模式顺序，详情请参见图 1 中的实现步骤。

1. 应用已上电且工作于运行模式：STPMIC1BPQR 处于 Power-ON-main 状态。
2. 如果请求了 LPLV\_Stop 工作模式，则软件准备 LPLV\_ 进入流程：
  - a. 进行 STM32MP15x 等设置：停止一些时钟，将 DDR 设为自刷新，设置 PWRLP\_TEMPO。
  - b. STPMIC1BPQR 设置：
    - BUCK1 (VDDCORE)：1.2 V HP 处于主模式，0.9 V LP 处于备用模式
    - BUCK2 (VDD2\_DDR)、BUCK3 (VDD)：ON HP 处于主模式，而 ON LP 处于备用模式。
3. STM32MP15x 设置 PWR\_CR1 寄存器的 LPDS 和 LVDS 位，等待进入 LPLV-：PWR\_ON 信号被置为无效，而 STM32MP15x 进入 LPLV-stop 模式。STPMIC1BPQR 进入 Power-ON-alternate 状态：
  - Buck1 电压下降到 0.9 V
  - 所有降压稳压器进入低功耗模式。
4. 遇到唤醒事件时，STM32MP15x 退出 LPLV-模式，并使 PWR\_ON 信号有效：
  - a. STM32MP15x tSEL\_VDDCORETEMPO 超时，等待 VDDCORE 达到运行模式工作电压水平。

**注意**

tSEL\_VDDCORETEMPO = 380 μs 典型值（参见[5]和[6]）。

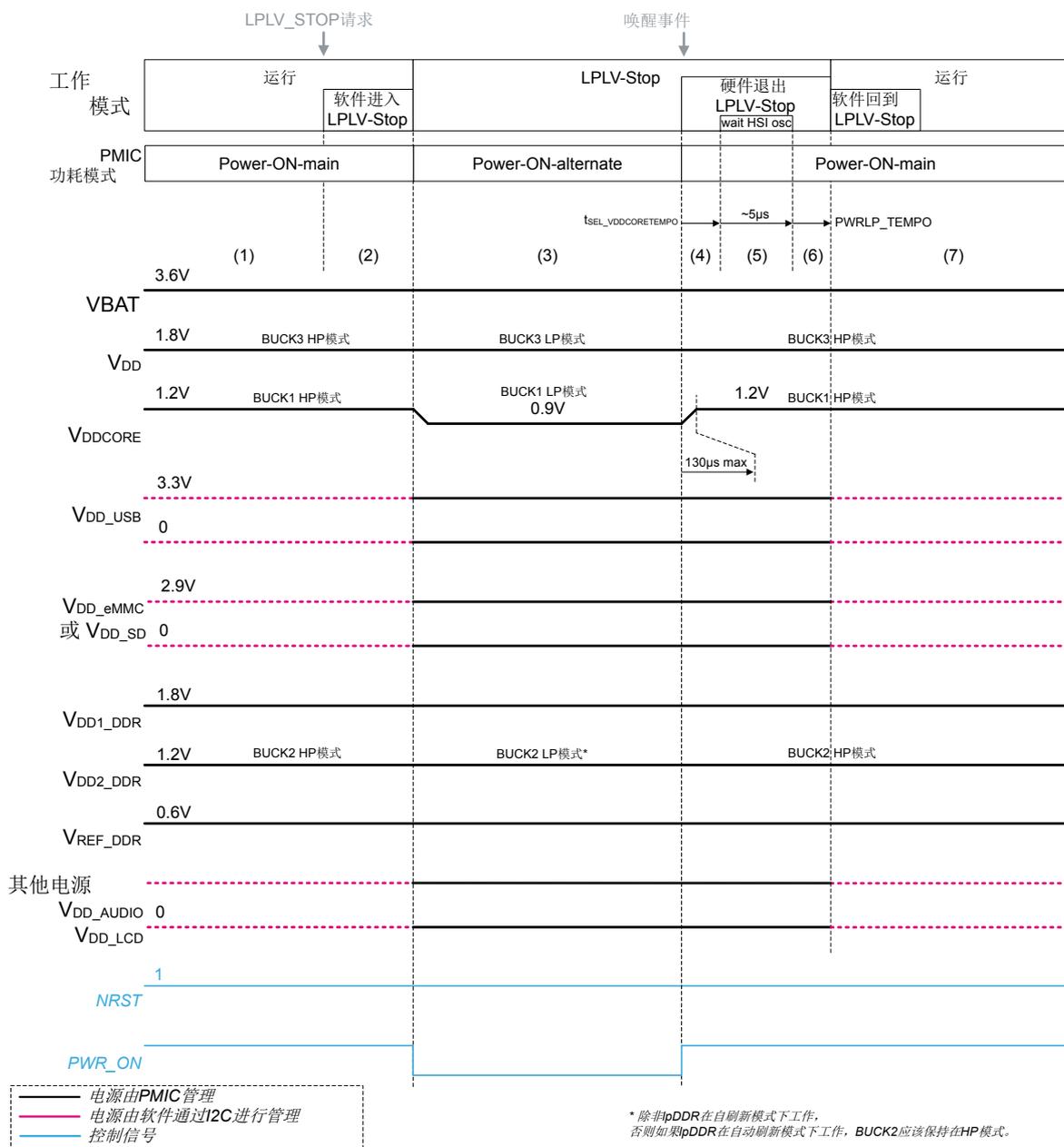
- b. STPMIC1BPQR 进入 Power-ON-main 状态：
  - BUCK1 电压从 0.9 V 上升到 1.2 V（不超过 130 μs）

**注意**

STPMIC1BPQR Buck1 具有 2.3 mV/μs 最低转换速率。

- 与此同时，所有降压转换器从低功耗变为高功耗模式 (100 μs)。

5. tSEL\_VDDCORETEMPO 延时过后，将在 STM32MP15x 中执行时钟恢复过程。
6. 一旦 STM32MP15x HSI 时钟振荡器逐步稳定 (~5 μs)，PWRLP\_TEMPO 定时器超时，等待外设逐步稳定。PWRLP\_TEMPO 是 STM32MP15x 专用定时器，设计目的是在应用从 LPLV 模式变为运行模式时，等待稳压器恢复。如果使用 STPMIC1，则 PWRLP\_TEMPO 延迟可以设为 0，或者旁路，因为 tSEL\_VDDCORETEMPO 持续时间超过 STPMIC1BPQR 整体恢复时间。然而，如果应用设为 LP\_（而不是 LPLV\_），则 STM32MP15x 不会等待 tSEL\_VDDCORETEMPO 延迟。所以，PWRLP\_TEMPO 必须设为 100 μs，以便让 STPMIC1BPQR 稳压器从 LP 模式恢复到 HP 模式。
7. 当 PWRLP\_TEMPO 延时过后，应用程序进入运行模式。软件从 LPLV-模式恢复（恢复时钟、从自刷新恢复 lpDDR...）。

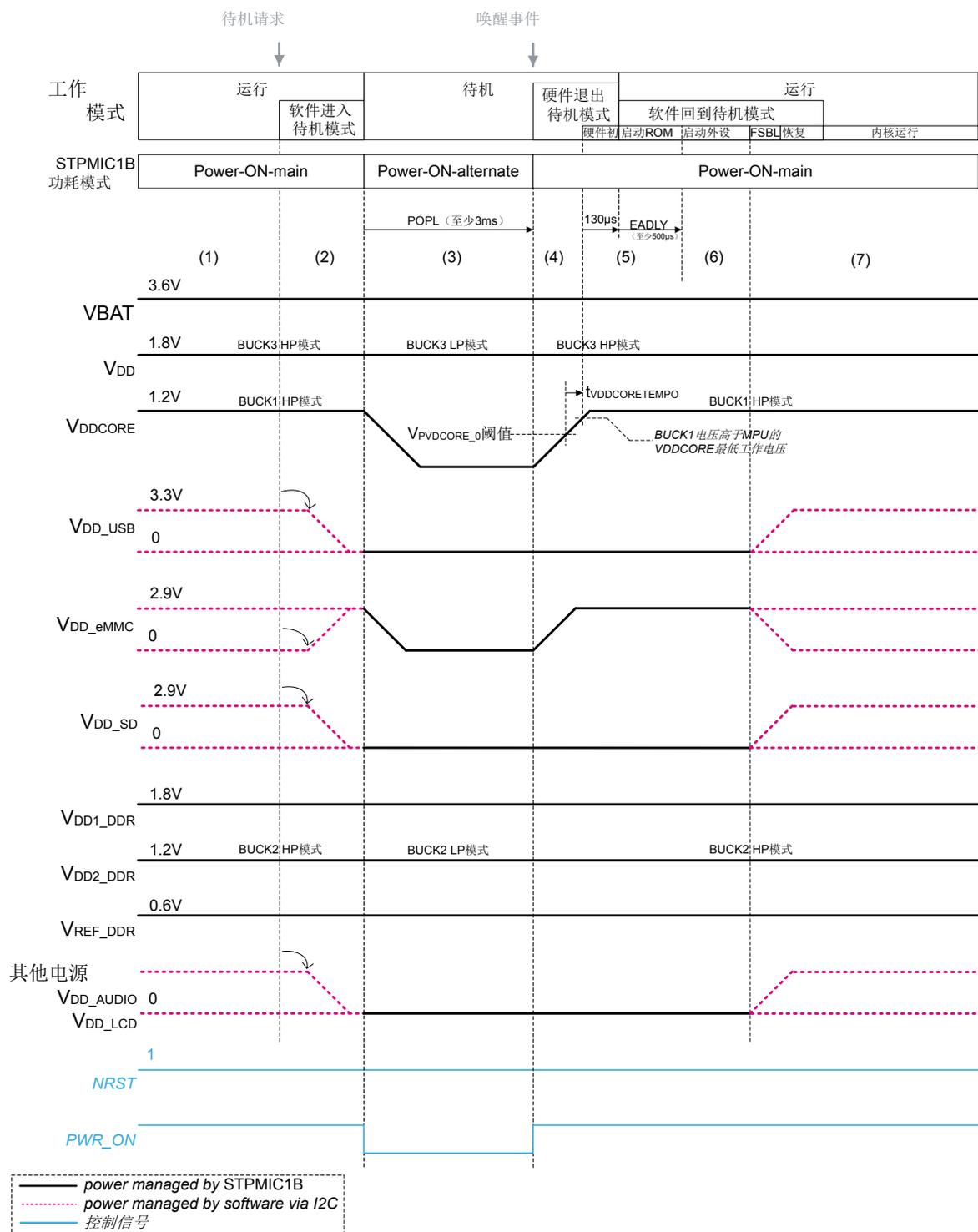
**图 6. LPLV-Stop 模式顺序**


### 5.3.2 待机模式

图 6 中显示应用待机模式顺序，详情请参见图 1 中的实现步骤。

在此应用中，退出待机模式后，eMMC Flash 存储器电压 (VDD\_eMMC) 必须存在，允许 STM32MP15x 读取启动软件 (FSBL)。在待机模式下，IpDDR 存储器处于自刷新状态。

- 应用程序上电并在运行模式下运行；STPMIC1BPQR 处于 Power-ON-main 状态。
- 请求待机模式之后，软件准备进入待机模式，方法是更改：
  - STM32MP15x 设置，比如：
    - 停止部分时钟
    - 将 DDR 设为自刷新
    - 设置 POPL 和 EADLY  
有关 POPL 的详细信息，请参考 POPL 定时器。  
有关 EADLY 定时器的详细信息，请参见 EADLY 定时器。在该应用中，VDD\_eMMC 在 500  $\mu$ s 之内上升 (LDO2)。所以，EADLY 可以设为 500  $\mu$ s 最小值，或者保持默认值 (10 ms)，前提是不必重点关注待机状态唤醒的持续时间。
    - 定时器等。
  - STPMIC1BPQR 设置：
    - BUCK1 (VDDCORE)、LDO2 (VDD\_eMMC)：在主模式下为 ON HP，在备用模式下关闭。
    - LDO4 (VDD\_USB)、LDO5 (VDD\_SD)、LDO1 (VDD\_AUDIO)、LDO6 (VDD\_LCD)：在主模式和备用模式下均关闭。
    - BUCK2 (VDD2\_DDR)、BUCK3 (VDD) 在主模式下为 ON HP，在备用模式下为 ON LP
- STM32MP15x 复位 LPDS 和 LVDS 位，在进入待机模式时等待；STM32MP15x 进入待机模式后，PWR\_ON 信号被置为无效：
  - POPL 定时器启动，防止 STM32MP15x 在 POPL 时间过去之前退出待机模式。
  - STPMIC1BPQR 进入 Power-ON-alternate 状态：
    - BUCK1 (VDDCORE)、LDO2 (VDD\_eMMC)：稳压器关闭
    - BUCK2 (VDD2\_DDR)、BUCK3 (VDD) 稳压器进入低功耗模式
- 遇到唤醒事件时，STM32MP15x 退出待机模式，并将 PWR\_ON 信号置为有效：
  - STPMIC1BPQR 进入 Power-ON-main 状态：
    - BUCK1 (VDDCORE)、LDO2 (VDD\_eMMC)：稳压器开启
    - BUCK2 (VDD2\_DDR)、BUCK3 (VDD) 稳压器进入高功耗模式
  - 一旦 VDDCORE 电压高于 VPVDCORE\_0 min 阈值，启动 tVDDCORETEMPO。只要 tVDDCORETEMPO 定时器没有结束，STM32MP15x 即保持内部复位状态。  
STM32MP15x 内部电压阈值，VPVDCORE\_0 上升沿为 0.95 V 最低值。  
STM32MP15x 内部延迟。tVDDCORETEMPO 为 200  $\mu$ s 最低值。
- 在 tVDDCORETEMPO 时延过后，STM32MP15x 退出内部复位状态 (VDDCORE\_OK)：
  - VDDCORE 电压高于 STM32MP15x VDDCORE 最低工作电压。  
BUCK1 具有 2.3 mV/ $\mu$ s 最低转换速率，确保 VDDCORE 电压高于 STM32MP15x VDDCORE 最低工作电压 (当 tVDDCORETEMPO 时延过后)。
  - STM32MP15x 执行内部硬件初始化 (允许在约 130  $\mu$ s 的时间内加载 HSI 和选项字节)，然后进入运行模式。
  - EADLY 延时定时器将启动。
- 当 EADLY 延时定时器过后，启动 ROM 开始访问外部外设 (Flash 存储器)，以加载和执行启动软件。默认情况下，当 EADLY 时间过后，VDD\_eMMC 电压稳定下来：
  - 从 eMMC 读取启动 ROM，验证和执行 FSBL。
  - 从这一步骤，软件可以通过 I<sup>2</sup>C 接口对 STPMIC1BPQR 进行编程，设置任意稳压器。
- 软件检测到“退出待机模式”，之后恢复并运行内核软件。

**图 7. 待机模式顺序**


## 5.4 用户复位和崩溃恢复管理

如第 4.3 节中所介绍的，STM32MP15x 和 STPMIC1BPQR 两者都有双向低电平有效复位引脚互连（参见图 1 NRST 信号）。

如果发生 STM32MP15x 崩溃（iwdg1\_out\_rst 或 iwdg2\_out\_rst watchdog 超时），NRST 信号上的 STM32MP15x 会产生一个复位脉冲。复位脉冲由 STPMIC1BPQR 捕获，立即触发断电重启时序：先是 STPMIC1BPQR 下电时序，然后是 STPMIC1BPQR 上电时序。

电源循环允许外设在发生崩溃后正确重启和复位，特别是对于没有复位输入信号的外设。对于 eMMC、NAND、NOR 和 SD-Card 等外设启动器件和 Flash 存储器件，特别建议进行电源循环。电源循环不再 STPMIC1BPQRBUCK3 (VDD) 上执行，因为后者需要在复位期间保持启用状态（参见第 5.1.2 节获取 STPMIC1BPQRmask\_reset 选项的详细信息）。

如果用户按下复位按钮，则 STPMIC1BPQR 执行相同的电源循环顺序。

### 5.4.1 故障恢复管理或用户复位顺序

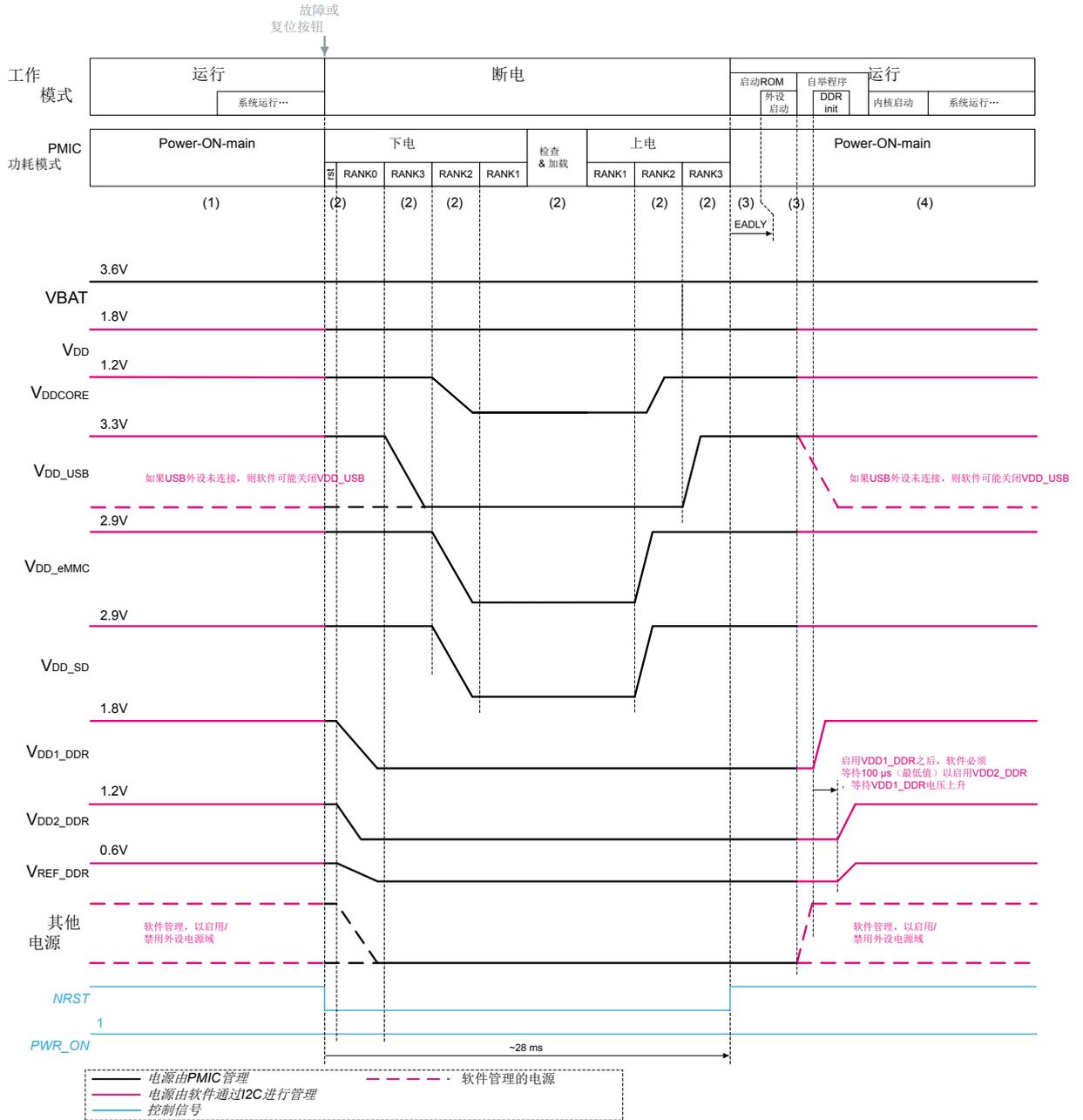
图 8 中显示的顺序说明了故障恢复顺序，详情请见图 1 中的实现步骤。在该顺序中，故障发生于运行模式期间（由 IWDG 复位处理）。但是，IWDG 复位可以在所有模式下发生，包括：运行模式、停止模式、LP-stop 模式、LPLV-stop 模式和待机模式。

1. 应用程序上电并在运行模式下运行；STPMIC1BPQR 处于 Power-ON-main 状态。发生崩溃（iwdg1\_out\_rst 或 iwdg2\_out\_rst 看门狗时间结束）或用户按下复位按钮，在 NRST 信号上产生脉冲。
2. STPMIC1BPQR 检测到复位有效确认（NRST 脉冲低电平），然后开始不可中断的断电重启：
  - a. STPMIC1BPQR 确认 NRST 低电平。
  - b. STPMIC1BPQR 执行下电时序。
  - c. STPMIC1BPQR 检查重新启动的条件（例如 VBAT、温度）并重新加载内部 NVM。
  - d. STPMIC1BPQR 执行上电时序。
  - e. STPMIC1BPQR 释放 NRST。  
如果复位信号 (NRST) 在此步骤中仍然确认有效（例如，用户仍然按住复位按钮），STPMIC1BPQR 将等待复位信号被释放，然后才会启动复位电路。这是为了避免 STPMIC1BPQR 重复断电重启回路。

注意

*STPMIC1BPQR 断电重启持续约 28 ms。*

3. NRST 信号上升，因为 STM32MP15x 和 STPMIC1BPQR 释放各自的相关复位引脚（且复位按钮被释放）：
  - a. STM32MP15xEADLY 延时定时器 (10 ms) 将启动。有关 EADLY 定时器的详细信息，请参见 EADLY 定时器。
  - b. 当 EADLY 延时过后，启动 ROM 开始访问外部外设（eMMC 或 SD-card，具体取决于 STM32MP15x 启动引脚设置），以加载和执行引导软件。
  - c. 自举程序可以控制任意 STPMIC1BPQR 稳压器（比如初始化 LCD 和绘制启动画面）。
4. 自举程序初始化 DDR，然后加载并执行内核：
  - a. LDO3 (VDD1\_DDR) 在旁路模式下被启用。软件等待至少 100  $\mu$ s。
  - b. DDR\_VREF (VREF\_DDR) 被启用。
  - c. BUCK2 (VDD2\_DDR) 被启用为 1.2 V。软件等待至少 1.4 ms，然后 BUCK2 就绪。
  - d. 软件初始化 STM32MP15xDDR 控制器和 IpDDR 器件。
  - e. 自举程序加载并执行内核，然后内核初始化。
  - f. 系统进入运行状态。

**图 8. 故障恢复顺序**


## 5.5 软件管理示例

本节介绍一种可能的 STPMIC1BPQR 管理软件集成策略（由 STM32MP15x 实施）。

OpenSTLinux 软件发行版集成了 STM32MP15x 驱动和配置 STPMIC1BPQR 稳压器的方式，以匹配第 5.1 节中的工作模式。

总结 OpenSTLinux 的主要集成点：

- 与 STPMIC1BPQR 的对接是在 FSBL（例如：TF-A）和 SSBL（例如：U-Boot）之间的启动链的低层“安全监控”部分执行的。
- 下面表 4 中介绍的 STPMIC1BPQR 电源管理策略是使用 Linux regulator framework-binding 术语在安全监控 DTS 文件（通常是 TF-A）中配置的。
- 每个 STPMIC1BPQR 电源被视为一个“调节器”，软件应用程序和驱动程序寄存器被视为“消费者”。通常情况下，在接到消费者请求时，再启用调节器。内核电源的例外情况是：无论消费者注册状态如何，内核电源都必须保持活跃，并保持“常开”。

如需详细了解 OpenSTLinux 中的电源管理功能，请参照以下在线用户指南文章：

[https://wiki.st.com/stm32mpu/wiki/Power\\_overview](https://wiki.st.com/stm32mpu/wiki/Power_overview)

[https://wiki.st.com/stm32mpu/wiki/Regulator\\_overview](https://wiki.st.com/stm32mpu/wiki/Regulator_overview)。

**表 4. STPMIC1BPQR 电源管理选项**

应用程序电源域	STPMIC1BPQR 电源	默认上电状态 (NVM)	功率状态			选项/注释
			STM32MP15x = 运行/停止 PMIC= POWER_ON Main	STM32MP15x = LPLV-stop PMIC= POWER_ON Alternate	STM32MP15x = 待机 DDR 关闭 PMIC= POWER_ON Alternate	
VDD	BUCK3	1.8V	1.8V / HP <sup>(1)</sup> / 常开 <sup>(2)</sup>	1.8V / LP	1.8V	Mask_reset <sup>(3)</sup> , 过电流保护 <sup>(4)</sup>
VDDCORE	BUCK1	1.2V	1.2V / HP / 常开	0.9V / LP	关闭	过电流保护
VDD_USB	LDO4	3.3V	3.3V / 消费者驱动 <sup>(5)</sup>	3.3V / 消费者驱动	关闭	-
VDD1_DDR	LDO3	OFF	1.8V / 常开	1.8V	关闭	过电流保护
VDD2_DDR	BUCK2	OFF	1.2V / HP / 常开	1.2V / LP	关闭	过电流保护
VREF_DDR	DDR_REF	OFF	0.6V / 常开	0.6V	关闭	-
VDD_eMMC	LDO2	2.9V	2.9V / 消费者驱动	2.9V / 消费者驱动	关闭 / 启动 <sup>(6)</sup>	-
VDD_SD	LDO5	2.9V	2.9V / 消费者驱动	2.9V / 消费者驱动	关闭	-
VBUS_DR	PWR_USB_SW	OFF	ON	关闭（MP1 侧不支持挂起）	关闭	纯软件。无备用寄存器
VDD_AUDIO	LDO1	OFF	1.8V / 消费者驱动	1.8V / 消费者驱动	关闭	-
VDD_LCD	LDO6	OFF	1.8V / 消费者驱动	1.8V / 消费者驱动	关闭	-

1. STPMIC1BPQR 稳压器的高功耗/低功耗模式。参见[2]
2. 常开：即使没有软件消费者，也让内核电压保持正常。
3. Mask\_reset：指定该调节器的 STPMIC1BPQR mask\_reset 选项，使其不受复位电源循环的影响。（参见第 5.1.2 节）
4. 过电流保护：指定该调节器的 STPMIC1BPQR 选项 OCP\_OFF。过电流检测导致 STPMIC1BPQR 关断。请参见 [2]
5. 消费者驱动：Linux 驱动程序根据消费者要求开启/关闭调节器。进入低功耗模式后，应用最后一个运行状态（主模式在备用模式下的复制）。在低功耗模式之前，用户必须考虑所需的状态。例如，当通电的外设为唤醒源时。
6. 启动：在进入低功耗模式（切换到 STPMIC1BPQR Alternate 模式）之前，软件必须将此调节器设为在 POWER\_ON 主模式下开启，以便在唤醒时立即通电（切换回主模式）。

## 6 USB 端口管理

### 6.1 USB 端口使用传统 micro-AB (OTG) 或 micro-B 插座 (DR)

图 1、图 2 和图 3 介绍了用于支持 USB OTG 扩展的 micro-AB 插座。或者，只有在期望 USB DR 模式时才使用 micro-B 插座（具有 ID 引脚）。使用 micro-B 插座时，应用原生支持 USB 外设模式；如果将“非 USB 兼容”适配器引脚 ID 设置为接地，则其也支持 USB 主机模式。

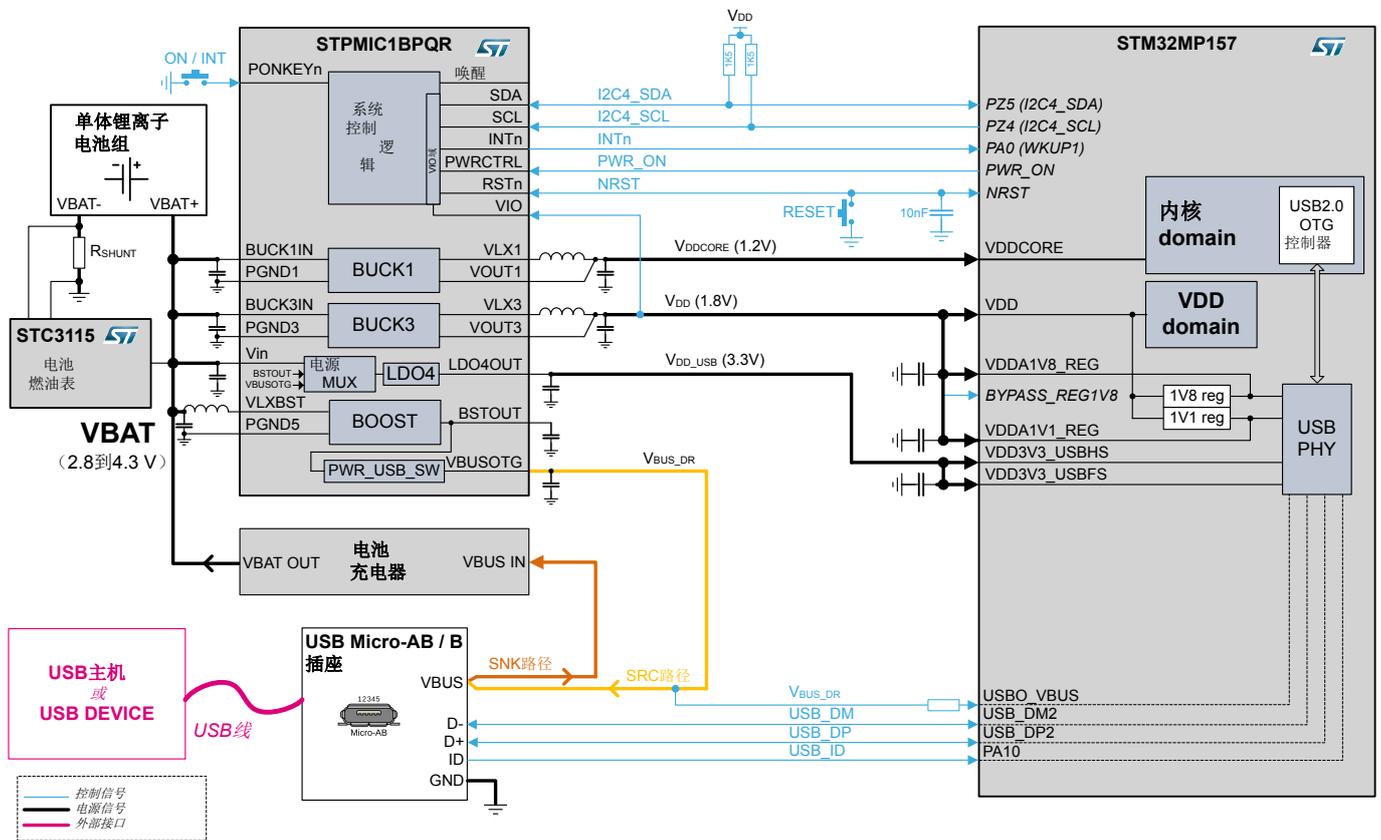
**注意** 适配器代表 USB micro-B 插头，其 ID 引脚接地到 USB type-A 插座。

图 9 详细介绍 USB 端口如何与 Type micro-B 或 Type micro-AB 插座集成

USB 插座的 VBUS 引脚连接到 VBUS\_DR 电源域，互连至：

- STPMIC1BPQR VBUSOTG 引脚：PWR\_USB\_SW 作为电源开关从升压 SMPS 转换器提供 VBUS 给 USB 插座（用于 A 会话），或者检测来自 USB 插座的有效 VBUS（用于 B 会话）。
- STM32MP15x USB0\_VBUS 引脚检测面向 B 会话的 VBUS\_DR 电压是否存在
- 电池充电器输入通过 USB 插座（处于 B 会话）为锂离子电池组充电。

图 9. 使用 USB micro-AB 或 micro-B 插座实现 USB 端口管理



**注意** 1. STM32MP15x 和 STPMIC1BPQR 是简化的图形，专注于 USB 功能集成。

2. STM32MP15x 解耦方案未显示（参见[1]相关章节）。

3. STPMIC1BPQR 分立器件值未显示（参见[2]）。

4. 没有显示附加保护，如 ESD、EMI 滤波和过电压等保护措施。

#### 6.1.1 USB 外设插头检测和移除检测管理

通过 USB 主机外设插头实现上电：

本文档假设图 9 中的应用程序处于断电模式，而 STPMIC1BPQR 处于关闭模式（参见表 3 获取关于工作模式的详细信息）。电池电压 VBAT 高于 STPMIC1BPQR VINOK\_rise，允许 STPMIC1BPQR 通电。来自 USB 主机外设或 USB 充电器外设的线缆插入 USB 插座，导致 VBUS\_DR 电压上升。由此将触发 STPMIC1BPQR 打开条件（STPMIC1BPQR VBUSOTG 引脚连接到 VBUS\_DR）。

USB 设备外设线缆插入不是 STPMIC1BPQR 通电条件，因为 USB\_ID 信号不是连接到 STPMIC1BPQR 以检测通电事件。

#### 通过 USB 外设插头实现唤醒：

本文档假设图 9 中的应用程序处于停止、LPLV-stop 或待机工作模式。默认情况下，STPMIC1BPQR 处于通电模式。

来自 USB 主机外设或 USB 充电器外设的线缆插入 USB 插座，导致 VBUS\_DR 电压上升。STPMIC1BPQR VBUSOTG 引脚上的上升电压通过 INTn 信号触发给 STM32MP15x 的中断，以唤醒 STM32MP15x。

STPMIC1BPQR 中断需要由软件在被激活之前启用，方法是设置 STPMIC1BPQR INT\_MASK\_R1[VBUSOTG\_RI] 位。详细信息，请参见[2]。

#### 通过 USB 外设移除实现唤醒：

本文档假设图 9 中的应用程序处于停止、LPLV-stop 或待机工作模式。默认情况下，STPMIC1BPQR 处于通电模式。

来自 USB 主机外设或 USB 充电器外设的线缆从 USB 插座移除，导致 VBUS\_DR 电压下降。STPMIC1BPQR VBUSOTG 引脚上的下降电压通过 INTn 信号触发给 STM32MP15x 的中断，以唤醒 STM32MP15x。

STPMIC1BPQR 中断需要由软件在被激活之前启用，方法是设置 STPMIC1BPQR INT\_MASK\_R1[VBUSOTG\_FA] 位。详细信息，请参见[2]。

#### 通过 USB 外设插头实现运行时检测：

图 9 中的应用程序处于运行工作模式。默认情况下，STPMIC1BPQR 处于通电模式。

- **USB 主机或充电器外设用例：**

来自 USB 主机外设或 USB 充电器外设的线缆插入 USB 插座，导致 VBUS\_DR 电压上升。在运行工作模式下，有两个检测源：

- STPMIC1BPQR VBUSOTG 引脚上的上升电压，通过 INTn 信号触发给 STM32MP15x 的中断。
- STM32MP15x USBO\_VBUS 引脚上的上升电压可以触发一个中断给 STM32MP15x USB OTG 控制器。为此，STM32MP15x USB OTG 控制器需要被启用。状态可以在 STM32MP15x GINTSTS[SRQINT] 中读取。详细信息，请参见[5]。

- **USB 设备外设用例：**

来自 USB 设备外设的线缆插入 USB 插座，将 USB\_ID 信号接地，将 STM32MP15x PA10 信号拉低。这将在 STM32MP15x USB OTG 控制器中触发一个中断。

为此，STM32MP15x USB OTG 控制器需要被启用。状态可以在 STM32MP15x GOTGINT [IDCHNG]中读取。详细信息，请参见[5]。

#### 通过 USB 外设移除实现运行时检测：

图 9 中的应用程序处于运行工作模式。默认情况下，STPMIC1BPQR 处于通电模式。

- **USB 主机或充电器外设用例：**

来自 USB 主机外设或 USB 充电器外设的线缆从 USB 插座移除，导致 VBUS\_DR 电压下降。在运行工作模式下，有两个检测源：

- STPMIC1BPQR VBUSOTG 引脚上的电压下降，通过 INTn 信号触发一个中断给 STM32MP15x。STM32MP15x USBO\_VBUS 引脚上的电压下降在 STM32MP15x USB OTG 控制器中触发一个中断。为此，STM32MP15x USB OTG 控制器需要被启用。状态可以在 STM32MP15x GINTSTS[DISCINT]中读取。详细信息，请参见[5]。

- **USB 设备外设用例：**

来自 USB 外设设备的线缆从 USB 插座移除，导致 USB\_ID 信号转为高电平，因此 STM32MP15x PA10 信号也转为高电平。这将在 STM32MP15x USB OTG 控制器中触发一个中断。

为此，STM32MP15x USB OTG 控制器需要被启用。状态可以在 STM32MP15x GOTGINT [IDCHNG]中读取。详细信息，请参见[5]。

## 6.1.2 在 USB 主机模式下从电池电压生成 VBUS\_DR 电压

当 USB 设备外设插入且被 STPMIC1BPQR 检测到（参见第 6.1.1 节），STPMIC1BPQR 发起 A 会话：

1. 软件在 A 设备模式（主机模式）下初始化 USB OTG 控制器

2. 软件对 STPMIC1BPQR 进行设置，从电池生成 VBUS\_DR：
  - a. 启用 STPMIC1BPQR 升压转换器（将 VBAT 转化为 5.2 V）
  - b. 等待至少 1.5ms，使 BSTOUT 电压稳定下来
  - c. 启用 STPMIC1BPQR PWR\_USB\_SW 电源开关：VBUS\_DR 上升到 5.2 V

相反，当 USB 设备外设被移除且分离状态被 STM32MP15x 检测到，它结束 A 会话并禁用 VBUS\_DR：

1. 禁用 STPMIC1BPQR PWR\_USB\_SW 电源开关：VBUS\_DR 被放电下拉功能拉低（参见第 6.1.3 节）
2. 禁用 STPMIC1BPQR 升压转换器

### 6.1.3 VBUS\_DR 放电下拉功能管理 STPMIC1BPQR

STPMIC1BPQR PWR\_USB\_SW 电源开关有一个可编程的下拉功能，用于对 VBUS\_DR 路径上的解耦电容进行放电。

在 STPMIC1BPQR 通电后，下拉功能需要通过软件启用（默认是禁用的）。

PWR\_USB\_SW 启用后（A 会话有效），下拉功能由 STPMIC1BPQR 自动禁用。当 USB 设备外设被移除，且分离信号被 STM32MP1 系列微处理器检测到，STM32MP1 系列微处理器会禁用 STPMIC1BPQR PWR\_USB\_SW。下拉功能将由 STPMIC1BPQR 自动启用，让 VBUS\_DR 变为低（对 VBUS\_DR 上的解耦电容器进行放电）。

当 USB 主机外设连接到 USB 插座，且 B 会话有效电压被检测到，软件会禁用 STPMIC1BPQR PWR\_USB\_SW 下拉功能，避免持续的电流消耗。当 USB 主机外设被移除，且分离信号被 STM32MP1 系列微处理器检测到，STPMIC1BPQR PWR\_USB\_SW 被软件设为下拉，对 VBUS\_DR 彻底放电。

### 6.1.4 STM32MP15x USB PHY 由放电后的电池（低于 3.3 V）供电

STM32MP15x USB PHY 拥有三个电压域：1.1 V、1.8 V 和 3.3 V（参见图 9）：

1.8 V (VDDA1V8\_REG) 由 VDD 电源（来自 STPMIC1BPQR BUCK3）供电。

1.1 V (VDDA1V1\_REG) 由 STM32MP15x 1V1\_reg（内部 STM32MP15xLDO，将 VDD 作为电源）供电。

3.3 V (VDD3V3\_USBHS 和 VDD3V3\_USBFS) 由 VDD\_USB（来自 STPMIC1BPQR LDO4）供电。

电池放电后（VBAT 电压低于 3.3 V）后，STPMIC1BPQR LDO4 不能生成从 VBAT 生成 3.3 V 的 VDD\_USB 电压（PMIC 的 VIN 引脚）。所以，电压太低，不能以 3.3 V 电压为 USB PHY 电压域供电。

为了从部分放电的电池以合适的电压（VDD\_USB，3.3 V）为 STM32MP15x USB PHY 供电，它使用 STPMIC1BPQR LDO4 输入之一：VIN、BSTOUT 或 VBUSOTG。STPMIC1BPQR 自动管理来自最高电压源的 LDO4 输入。

#### USB 主机外设用例

当 USB 主机外设（例如个人计算机）插入后，它提供 VBUS (~5 V) 到应用：VBUS\_DR（以及 STPMIC1BPQR VBUSOTG）电压上升。LDO4 输入自动切换到 VBUSOTG 电源，因为它高于电池电压：VBUSOTG = VBUS ~5 V 且 VBAT ≤ 3.3 V。

以 ~5 V 作为输入电源时，LDO4 能够生成电压为 3.3 V 的 VDD\_USB。

#### USB 设备外设用例

当 USB 设备外设（例如 USB 存储棒）插入且被 STM32MP15x 检测到（参见第 6.1.1 节），STM32MP15x 发起 A 会话，然后 STPMIC1BPQR 通过升压转换器从电池电压 (VBAT) 生成 VBUS\_DR 电压（参见章节第 6.1.2 节）。升压转换器输出 (BSTOUT) 生成电压为 5.2 V 的电源。LDO4 输入自动切换到 BSTOUT 电源，因为 BSTOUT 电压高于电池电压：BSTOUT = 5.2 V 且 VBAT ≤ 3.3 V。

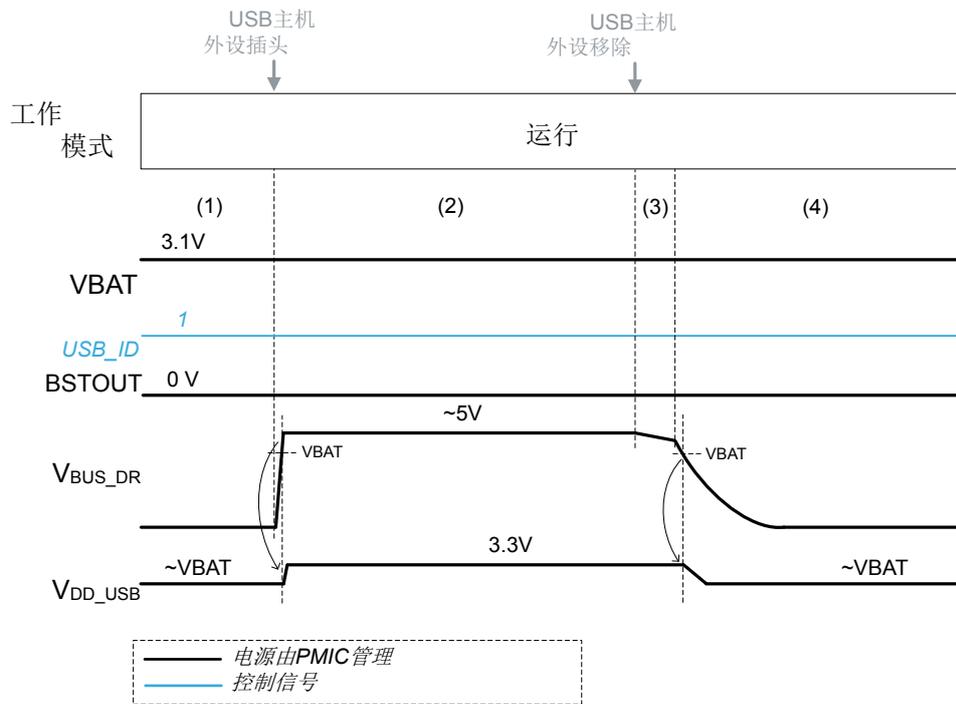
以 5.2 V 作为输入电源时，LDO4 能够生成电压为 3.3 V 的 VDD\_USB。

### 6.1.5 USB 主机外设电源管理总结

图 10 中的时序专门用于 USB 主机外设管理并根据第 6.1 节中所示的实现提供从第 6.1.1 节到第 6.1.4 节的总结。为了简单起见，假定电池电压在整个时序中保持不变（电池放电未显示）。例如，这可以是一台个人计算机。

1. 应用由放电后的电池供电 (VBAT = 3.1 V) 并且在运行工作模式下工作。STPMIC1BPQR 处于 Power-ON-main 状态。STPMIC1BPQR PWR\_USB\_SW 放电下拉功能最初由软件启用（参见第 6.1.3 节）

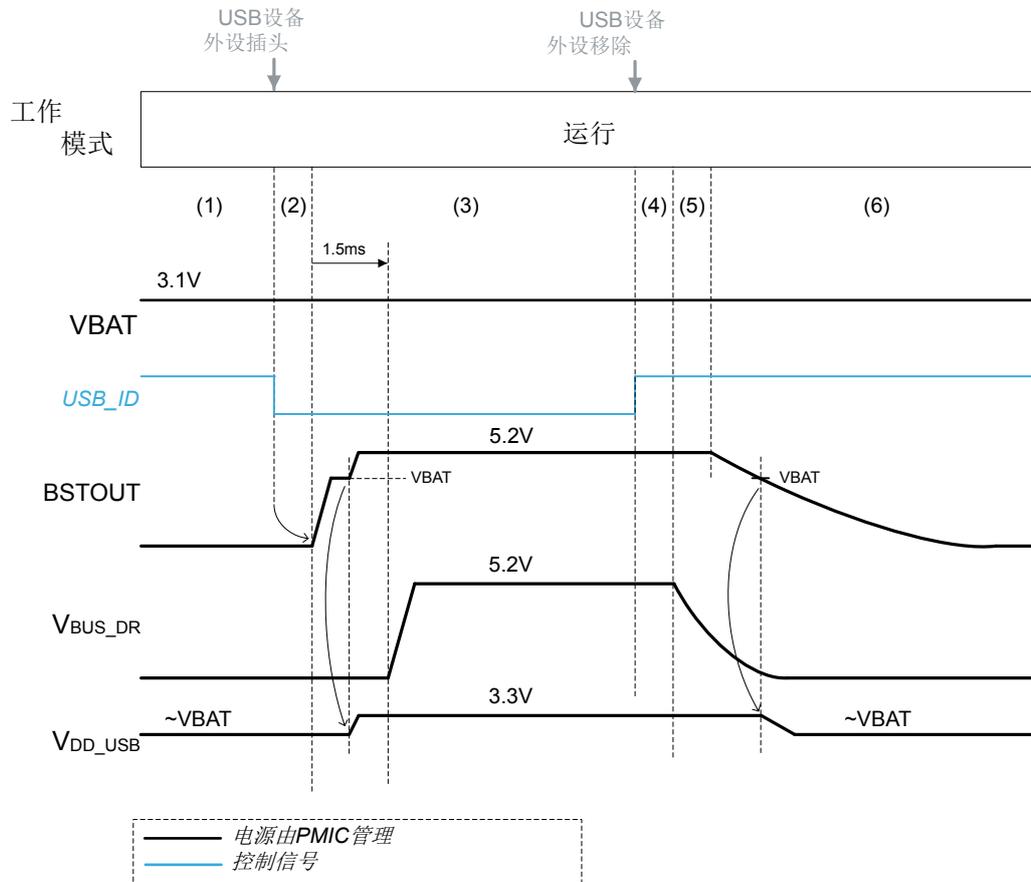
2. USB 主机外设插入 USB 插座：
  - a. VBUS\_DR 上升（因为 VBUS 电压由主机外设提供），使 STPMIC1BPQR VBUSOTG 和 STM32MP15x USB0\_VBUS 提高。两种信号均可以触发中断（参见第 6.1.1 节）。
  - b. 一旦中断被软件捕获，软件在 B 设备模式（设备模式）下初始化 STM32MP15xUSB OTG 控制器并发起 USB 通信。
  - c. 一旦 VBUS\_DR 高于 VBAT，STPMIC1BPQR LDO4 由 VBUS\_DR 供电（参见第 6.1.4 节），而 LDO4 输出 (VDD\_USB) 恢复为 3.3 V。
  - d. 软件禁用 STPMIC1BPQR PWR\_USB\_SW 放电下拉功能（参见第 6.1.3 节）
3. USB 主机外设从 USB 插座移除：
  - a. VBUS\_DR 缓慢下降。
  - b. 一旦 VBUS\_DR 低于定义的电压阈值，STPMIC1BPQR PWR\_USB\_SW（通过 VBUSOTG 引脚）或 STM32MP15xUSB OTG 控制器（通过 USB0\_VBUS 引脚）触发中断（参见第 6.1.1 节）。
  - c. 软件终止 USB B 设备会话。
4. 软件启用 STPMIC1BPQR PWR\_USB\_SW 放电下拉功能：
  - a. VBUS\_DR 变低。
  - b. 一旦 VBUS\_DR 低于 VBAT，STPMIC1BPQR LDO4 由 VBAT 供电，而 LDO4 输出 (VDD\_USB) 低于 3.3 V（参见第 6.1.4 节）。

**图 10. USB 主机外设电源管理总结**


## 6.1.6 USB 设备外设电源管理总结

图 11 中的时序专门用于 USB 设备外设管理，并根据第 6.1 节中所示的实现提供从第 6.1.1 节到第 6.1.4 节的总结。为了简单起见，假定电池电压在整个时序中保持不变（电池放电未显示）。例如，这可以是一个 USB 大容量存储加密狗。

1. 应用由放电后的电池供电 (VBAT = 3.1 V) 并且在运行工作模式下工作。STPMIC1BPQR 处于 Power-ON-main 状态。
2. USB 设备外设插入 USB 插座：
  - a. 当 USB 线缆插入 USB 插座后，USB\_ID 信号接地。当 STM32MP1 系列微处理器 PA10 信号 (USB\_ID) 变成低电平，将触发一个中断到 STM32MP1 系列微处理器 USB OTG 控制器（参见第 6.1.1 节）。
  - b. 软件在 A 设备模式（主机模式）下初始化 STM32MP1 系列微处理器 USB OTG 控制器
3. 软件启用 STPMIC1BPQR 升压转换器：
  - a. 一旦 BSTOUT 升压输出电压高于 VBAT，STPMIC1BPQR LDO4 由 BSTOUT 供电（参见第 6.1.4 节）而 LDO4 输出 (VDD\_USB) 恢复为 3.3 V。
  - b. 在最短 1.5 ms（参见第 6.1.2 节）的延时过后，软件启用 STPMIC1BPQR PWR\_USB\_SW 电源开关：VBUS\_DR 电压上升。USB 设备外设通电。USB OTG 控制器检测到 USB 设备的存在，然后初始化 USB A 设备会话。
4. USB 设备外设从 USB 插座移除：
  - a. USB\_ID 信号被释放并转为高电平。当 STM32MP1 系列微处理器 PA10 信号 (USB\_ID) 转为高电平后，它触发一个中断到 STM32MP1 系列微处理器 USB OTG 控制器（参见第 6.1.1 节）。
  - b. 当 STM32MP1 系列微处理器 USB OTG 控制器检测到设备被移除，它会结束 USB A 设备会话。
5. 软件禁用 STPMIC1BPQR PWR\_USB\_SW 电源开关。PWR\_USB\_SW 的内部下拉由 STPMIC1BPQR 自动启用，导致 VBUS\_DR 被拉低：VBUS\_DR 下降（参见第 6.1.3 节）
6. 当 VBUS\_DR 放电时，软件禁用 STPMIC1BPQR 升压转换器：
  - a. BSTOUT 电压下降
  - b. 一旦 BSTOUT 升压输出电压低于 VBAT，STPMIC1BPQRLDO4 由 VBAT 供电，而 LDO4 输出 (VDD\_USB) 低于 3.3 V（参见第 6.1.4 节）。

**图 11. USB 设备外设电源管理总结**


### 6.1.7 通过 USB micro-B 插座应用实现电池充电和电池监控

本节将介绍图 9。其中重点论述，当充电器连接到 USB Type micro-B 插座时，STM32MP15x、STPMIC1BPQR，以及电池充电器之间的主要交互。

**注意** 电池充电算法和电池充电安全不在本应用笔记的讨论范围之内。

图 9 中的应用原生支持单体电池或并联单元 3.6 V 锂离子或锂聚合物电池。得益于 STPMIC1BPQR BOOST SMPS，也支持所谓的低截止电池。其他电池类型也可以支持，但仅限于 STPMIC1BPQR 最低和最高工作电压。

在此应用中，USB 充电器是一个通用 IC。它需要同时支持：

- 硬件充电：无电电池自主充电。
- 软件充电：充电过程和充电监测由 STM32MP15x 软件算法管理

如需了解以下功能，请参阅相应章节：

- 电池接口和存在检测请参见第 6.3.1 节。
- 电池监测请参见第 6.3.2 节。
- 如需了解关于从‘电量耗尽’模式进行上电和下电，以及电池充电的相关信息，请参见第 6.3.3 节。

## 6.2 USB 端口（使用 USB Type-C®插座）

图 12 提供关于 USB Type-C®端口与 USB Type-C®插座集成的详细信息。该应用示例支持：

- USB 2.0 高速接口
- 供电 (SRC) 角色：以最高 500 mA 的电流为附设的受电外设供电，并以‘下行面向端口’数据角色运行（USB 主机模式）。
- 受电 (SNK) 角色：从附设的供电外设获取能量（根据供电外设和电池充电器 IC 的能力，通常以最高 5 V / 3 A 为电池充电）并以‘上行面向端口’数据角色运行（USB 设备模式）。
- 不支持 USB Power-Delivery（包括双角色电源）

USB Type-C®控制器 (STUSB1600) 管理外设插头检测和附设识别它支持“电池耗尽”模式 (电池完全放电后进行充电) 或在外设附设之后唤醒应用程序。USB Type-C®控制器被配置为管理供电电源角色或受电电源角色。

供电外设附设之后, 应用作为受电端口运行, 扮演‘上行面向端口’数据角色 (USB 设备模式)。

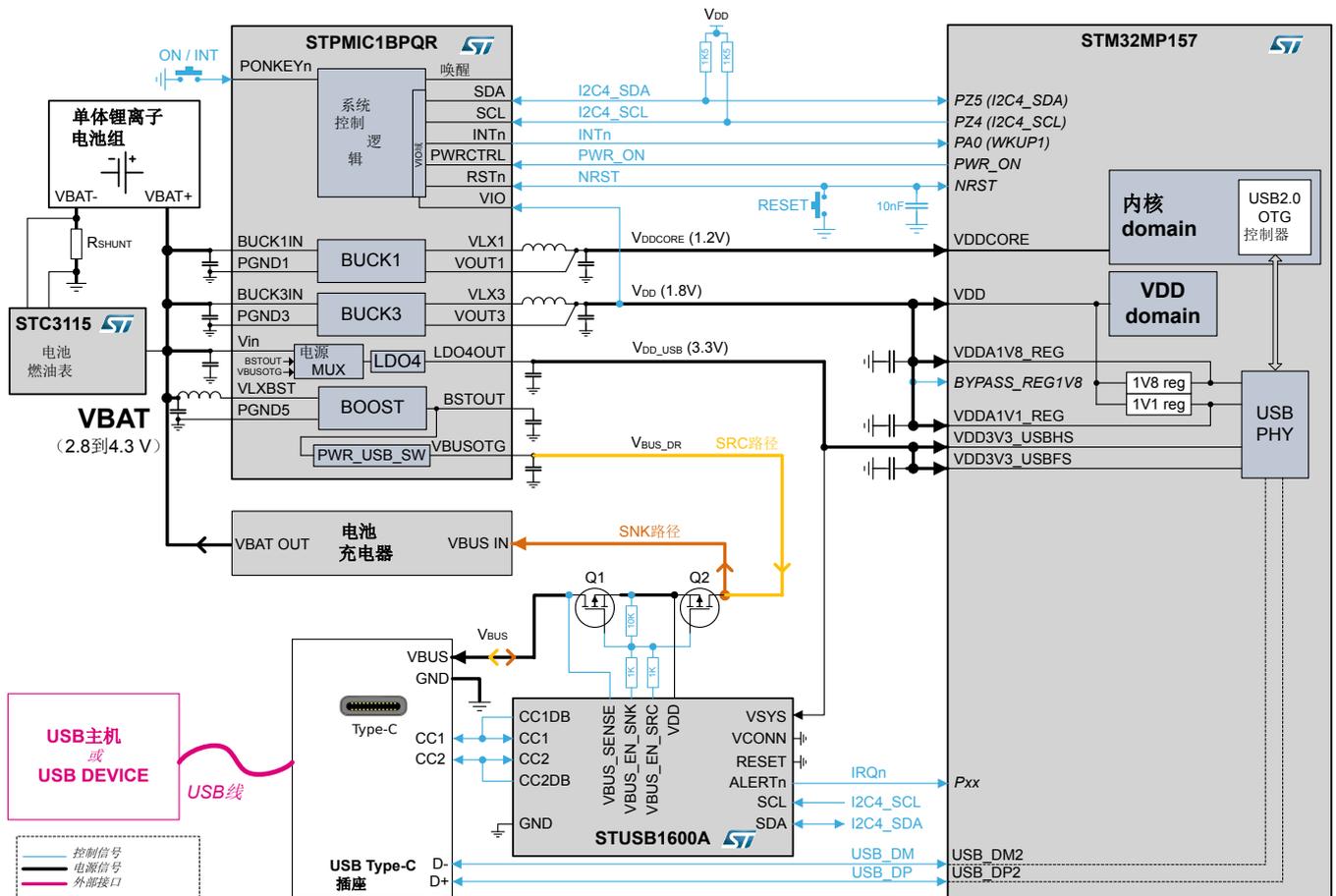
受电外设附设之后, 应用作为供电端口运行, 扮演‘下行面向端口’数据角色 (USB 主机模式)。

应用能够通过 STPMIC1BPQR 升压转换器和 PWR\_USB\_SW 电源开关提供最高 2.5 W (5 V / 500 mA) 电源。因此, 当受电外设附设时, 应用以“默认 USB 电源”广播受电外设。在连接时, 受电外设必须遵循 USB 2.0 规范, 以允许从应用接收 500 mA 电源。

USB Type-C®插座的 VBUS 引脚连接到双向电源开关 (包含两个采取背靠背配置的 P 沟道功率 Mosfet)。双向电源开关由 USB Type-C®控制器 (STUSB1600A) 控制, 用于管理: 供电或受电电源路径。

- 供电电源路径: 电池电压 VBAT 通过 STPMIC1BPQR 升压转换器转换为 5.2 V (从 VLXBST 到 BSTOUT)。BSTOUT 内部连接到 PWR\_USB\_SW 电源开关输入。PWR\_USB\_SW 输出 (VBUSOTG) 连接到 VBUS\_DR, 提供电压给双向电源开关 (旁路 USB Type-C®插座的 VBUS\_DR 到 VBUS)。
- 受电电源路径: USB Type-C®提供 VBUS 电压给双向电源开关 (将 VBUS 从 USB Type-C®旁路到 VBUS\_DR)。VBUS\_DR 传输到电池充电器用于电池充电, 以及传输到 STPMIC1BPQR VBUSOTG 引脚 (PWR\_USB\_SW) 用于 VBUS 检测 (为应用上电)。

图 12. 使用 USB Type-C®插座进行 USB 端口管理



注意

1. STM32MP15x、STPMIC1BPQR 和 USB Type-C®控制器 (STUSB1600A) 是简化的图形, 专注于 USB 功能集成。
2. STM32MP15x 解耦方案未显示 (参见[1]相关章节)。
3. STPMIC1BPQR 分立器件值未显示 (参见[2])。
4. 没有显示附加保护, 如 ESD、EMI 滤波和过电压等保护措施。

## 6.2.1 USB 外设插头检测和移除检测管理

通过 USB 供电外设插头实现上电（“电池耗尽”模式下的供电方-受电方连接）：

图 12 中的应用程序处于断电模式，且 STPMIC1BPQR 处于关闭模式。电池电压 VBAT 高于 STPMIC1BPQR VINOK\_rise，允许 STPMIC1BPQR 通电。

注意

参见表 3 获取关于工作模式的详细信息。

供电外设连接到 USB Type-C®插座。USB Type-C®控制器 (STUSB1600A) 将其 CC 引脚上的下拉终端置为有效，并广播为受电方（支持电量耗尽的电池）。供电外设检测下拉终端，建立供电方-受电方连接，并提供 VBUS。一旦 USB Type-C®控制器检测到有效电压（VBUS\_SENSE 引脚），它将闭合电源开关 (Q1/Q2)，使 VBUS\_DR 电压上升。由此将触发 STPMIC1BPQR 打开条件（STPMIC1BPQR VBUSOTG 引脚连接到 VBUS\_DR）。

USB 受电外设连接（供电方-受电方连接）不是应用程序的打开条件：当应用程序处于断电模式且 STPMIC1BPQR 处于关闭模式时，USB Type-C®控制器没有通电，无法检测附件，因为它没有能量来确认其 CC 引脚上的上拉终端 (Rp)。

通过 USB 供电外设插头实现唤醒：

图 12 中的应用停止、LPLV-stop 或待机工作模式。默认情况下，STPMIC1BPQR 处于通电模式。USB Type-C®控制器由处于“受电角色”的 STM32MP15x 完成初始化（参见[7]获取详细信息）。

供电外设连接到 USB Type-C®插座。因为 USB Type-C®控制器广播为受电方（USB Type-C®控制器确认其 CC 引脚上的下拉终端 (Rd)），供电外设检测下拉终端，建立供电方-受电方连接，并提供 VBUS。一旦 USB Type-C®控制器检测到有效电压（来自 VBUS\_SENSE 引脚），它闭合电源开关 (Q1/Q2)，使 VBUS\_DR 电压上升。有两个检测源可以唤醒 STM32MP15x：

- USB Type-C®控制器检测到的一个附设通过 IRQn 信号触发一个给 STM32MP15x 的中断。
- STPMIC1BPQRVBUSOTG 引脚上的上升电压通过 INTn 信号触发给 STM32MP15x 的中断。

通过 USB 受电外设插头实现唤醒：

图 12 中的应用停止、LPLV-stop 或待机工作模式。默认情况下，STPMIC1BPQR 处于通电模式。USB Type-C®控制器由处于“供电角色”的 STM32MP15x 完成初始化（参见[7]获取详细信息）。

受电外设连接到 USB Type-C®插座。因为 USB Type-C®控制器广播为供电方（USB Type-C®控制器确认其 CC 引脚上的上拉终端 (Rd)），它检测附设的受电方，然后通过 IRQn 信号触发给 STM32MP15x 的中断（唤醒 STM32MP15x）。

参见第 6.2.2 节获取关于 VBUS 电压生成的详细信息。

通过 USB 供电外设移除实现唤醒：

图 12 中的应用停止、LPLV-stop 或待机工作模式。默认情况下，STPMIC1BPQR 处于通电模式。USB 供电外设连接到 USB Type-C®插座，应用程序作为“Attached.SNK”（参见[9]）

供电外设从 USB Type-C®插座移除：

- USB Type-C®控制器检测外设的移除。USB Type-C®控制器断开电源开关 (Q1/Q2) 并通过 IRQn 触发给 STM32MP15x 的中断（唤醒 STM32MP15x）。一旦 STM32MP15x 被唤醒，它读取中断状态并发送给 USB Type-C®控制器，检测 USB 供电方的移除。
- VBUS\_DR 电压下降导致电压降到 STPMIC1BPQRVBUSOTG 引脚，通过 INTn 信号触发给 STM32MP15x 的中断（唤醒 STM32MP15x）。

通过 USB 受电外设移除实现唤醒：

图 12 中的应用停止、LPLV-stop 或待机工作模式。默认情况下，STPMIC1BPQR 处于通电模式。USB 受电外设连接到 USB Type-C®插座，应用程序作为“Attached.SRC”（参见[9]）

受电外设从 USB Type-C®插座移除。USB Type-C®控制器检测外设的移除。USB Type-C®控制器断开电源开关 (Q1/Q2) 并通过 IRQn 触发给 STM32MP15x 的中断（唤醒 STM32MP15x）。一旦 STM32MP15x 被唤醒，它将读取 USB Type-C®控制器中的中断状态，检测 USB 受电方的移除。STM32MP15x 禁用 STPMIC1BPQR PWR\_USB\_SW，然后是 STPMIC1BPQR 升压转换器。

为了将 STM32MP15x 从待机状态唤醒，来自 STPMIC1BPQR 或 USB Type-C®控制器的中断信号必须连接到 STM32MP15x 的一个唤醒源 (WKUPx) 输入。

### 运行时检测是否已附设 USB 受电外设或 USB 供电外设：

图 12 中的应用程序处于运行工作模式。默示情况下，STPMIC1BPQR 处于通电模式。在运行时，受电或供电 USB 外设插头或移除使用与前面描述的唤醒流程相同的检测机制。

## 6.2.2 在供电模式下从电池电压生成 VBUS 电压

当 USB 受电外设插入且连接被 USB Type-C 控制器（参见第 6.2.1 节）检测到后，它通过 IRQn 触发一个中断给 STM32MP15x：

1. STM32MP15x 软件读取中断寄存器的状态，并检测 USB 受电外设连接。
2. STM32MP15x 软件对 STPMIC1BPQR 进行设置，从电池生成 VBUS\_DR：
  - a. 启用 STPMIC1BPQR 升压转换器（将 VBAT 转换为 5.2 V）。
  - b. 等待至少 1.5ms 的延迟时间，使 BSTOUT 电压稳定下来（建议不少于 5ms）。
  - c. 启用 STPMIC1BPQR PWR\_USB\_SW 电源开关：VBUS\_DR 上升到 5.2 V。
3. USB Type-C 控制器闭合电源开关 (Q1/Q2)，使 VBUS 上升。应用处于“Attached.SRC”状态（参见[9]）。
4. STM32MP15x 软件可以发起面向下行的端口会话（USB 主机模式）。

## 6.2.3 STM32MP15x USB PHY 由放电后电压低于 3.3 V 的电池供电

参见第 6.2.4 节获取关于 STM32MP15x USB PHY 电源约束的概述。本节指的是图 12 中的应用。

### USB 供电外设用例：

当 USB 供电外设连接到应用时，最初设为“受电角色”的 USB Type-C®控制器作广播为受电外设，确保供电外设提供 VBUS (~ 5 V)。如果 VBUS 电压在有效范围内（参见[7]），USB Type-C®控制器闭合电源开关 (Q1/Q2)：VBUS\_DR (以及 STPMIC1BPQRVBUSOTG) 电压上升。LDO4 输入自动切换到 VBUSOTG 电源，因为 VBUSOTG 电压高于电池电压（参见[2]）：

- VBUSOTG = VBUS ~5 V
- VBAT ≤ 3.3 V.

以~5 V 作为输入电源时，LDO4 能够生成电压为 3.3 V 的 VDD\_USB。

### USB 受电外设用例：

当 USB 受电外设连接到应用后，应用通过升压转换器将电池电压转换为 VBUS 电压，从而为受电外设生成 VBUS 电压（参见第 6.2.2 节获取详细信息）：升压转换器输出 (BSTOUT) 生成 5.2 V 电源。LDO4 输入自动切换到 BSTOUT 电源，因为 BSTOUT 电压高于电池电压：

- BSTOUT = 5.2 V
- VBAT ≤ 3.3 V.

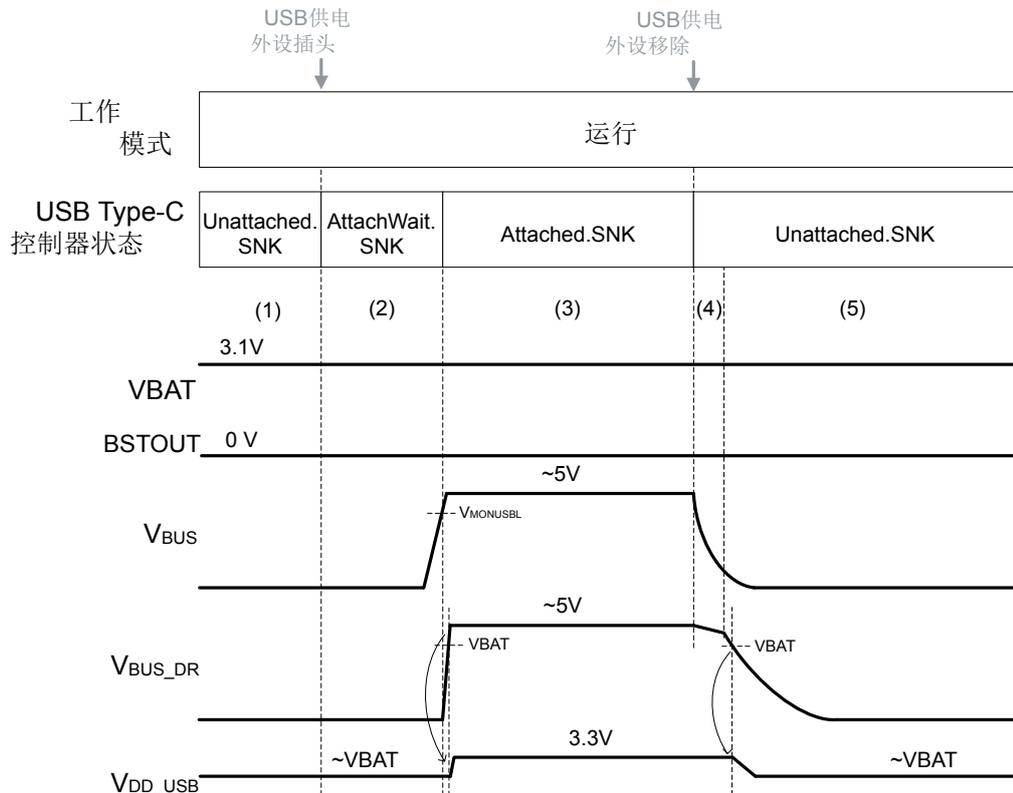
以 5.2 V 作为输入电源时，LDO4 能够生成电压为 3.3 V 的 VDD\_USB。

## 6.2.4 USB 供电外设电源管理总结

图 13 中的时序专门用于 USB 供电外设管理（如个人电脑主机）并根据图 12 中所示的实现提供第 6.2.1 节到第 6.2.3 节的总结。在此过程中，假定电池电压在整个时序中保持不变（电池充电未显示）。

1. 应用由放电后的电池供电 (VBAT = 3.1 V) 并且在运行工作模式下工作。STPMIC1BPQR 处于 Power-ON-main 状态。USB Type-C®控制器最初由 STM32MP15x 设置为“受电角色”并且在 unattached.SNK 状态下执行（参见[7]获取详细信息）。
2. USB 供电外设插入 USB 插座：
  - a. 因为 USB Type-C®控制器广播为 Unattached.SNK，它检测供电外设连接情况，然后进入 AttachWait.SNK 状态。
  - b. 一旦 USB 供电外设检测到与应用程序的连接，则其为 VBUS 供电。

3. 一旦 VBUS 高于 VMONUSBL 阈值（参见[7]获取详细信息）：
  - a. USB Type-C®控制器进入 Attached.SNK 状态，它闭合电源开关 (Q1/Q2) 并且触发一个中断给 STM32MP15x（参见第 6.2.1 节）。
  - b. VBUS\_DR 上升导致 STPMIC1BPQR VBUSOTG 上升，触发一个中断给 STM32MP15x（参见第 6.2.1 节）。
  - c. 一旦中断被软件捕获，软件在 B 设备模式（设备模式）下初始化 STM32MP15xUSB OTG 控制器并发起 USB 通信。
  - d. 一旦 VBUS\_DR 高于 VBAT，STPMIC1BPQR LDO4 由 VBUS\_DR 供电（参见第 6.2.2 节），而 LDO4 输出 (VDD\_USB) 恢复为 3.3 V。
  - e. 软件禁用 STPMIC1BPQR PWR\_USB\_SW 放电下拉功能（参见第 6.1.3 节）。
4. USB 供电外设从 USB 插座移除。一旦 USB Type-C®控制器检测到附设的外设（参见[7]获取详细信息）：
  - a. USB Type-C®控制器断开电源开关 (Q1/Q2)，它触发一个中断给 STM32MP15x，激活 VBUS 放电路径，然后进入 Unattached.SNK 状态。
  - b. 一旦 VBUS 放电路径被激活，VBUS 电压变低。
  - c. 一旦电源开关 (Q1/Q2) 断开，VBUS\_DR 开始缓慢放电。
5. 一旦来自 USB Type-C®控制器的中断被软件捕获：
  - a. 软件终止 USB B 设备会话。
  - b. 软件启用 STPMIC1BPQR PWR\_USB\_SW 放电下拉功能，将 VBUS\_DR 拉低。
  - c. 一旦 VBUS\_DR 低于 VBAT，STPMIC1BPQR LDO4 由 VBAT 供电，而 LDO4 输出 (VDD\_USB) 再次低于 3.3 V（参见第 6.2.3 节）。

**图 13. USB 供电外设电源管理总结**


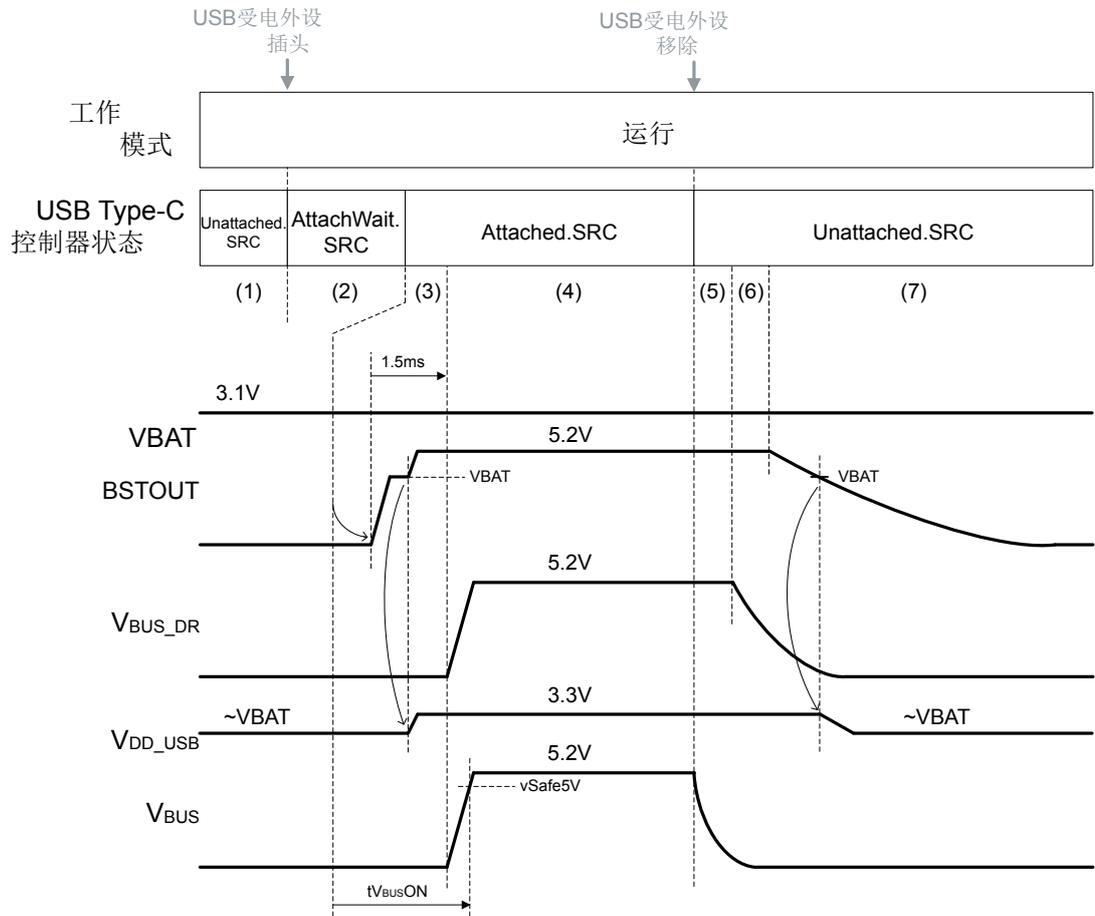
## 6.2.5 USB 受电外设供电管理总结

图 14 中的时序专门用于 USB 受电外设管理（如 USB 大容量存储加密狗）并根据图 12 中所示的实现提供第 6.2.1 节到第 6.2.3 节的总结。在此过程中，假定电池电压在整个时序中保持不变（电池放电未显示）。

1. 应用由放电后的电池供电 (VBAT = 3.1 V) 并且在运行工作模式下工作。STPMIC1BPQR 处于 Power-ON-main 状态。USB Type-C®控制器最初由 STM32MP15x 设置为“受电角色”并且在 Unattached.SRC 状态下执行（参见 [7] 获取详细信息）。
2. USB 受电外设插入 USB 插座。因为 USB Type-C®控制器广播为 Unattached.SRC:
  - a. USB Type-C®控制器检测受电外设连接。
  - b. USB Type-C®控制器进入 AttachWait.SNK 状态。
  - c. 经过一段去抖动时间后，为保证受电外设连接（参见 [9] 中的 tCCDebounce），USB Type-C®控制器进入 Attached.SRC 状态，它闭合电源开关 (Q1/Q2) 并触发一个中断给 STM32MP15x。

*注意* 当 USB Type-C®控制器进入 Attached.SRC 状态，VBUS 电压必须在 tVBUSON 延时（275ms 最高值）之内高于 vSafe5V 电压。详细信息，请参见 [9]。

3. 一旦中断被 STM32MP15x 软件捕获：
  - a. 软件初始化 STM32MP15x USB OTG 控制器，处于 A 设备模式（主机模式）。
  - b. 软件启用 STPMIC1BPQR 升压转换器并将定时器的持续时间设为 1.5 ms 最小值（参见第 6.2.3 节）
  - c. 一旦 BSTOUT 升压输出电压高于 VBAT，STPMIC1BPQR LDO4 由 BSTOUT 供电，而 LDO4 输出 (VDD\_USB) 恢复为 3.3 V。
4. 一旦 1.5ms 定时器延时过去，软件启用 STPMIC1BPQR PWR\_USB\_SW 电源开关：
  - a. VBUS\_DR 电压上升导致 VBUS 上升。
  - b. USB 设备外设通电。USB OTG 控制器检测到 USB 受电外设的存在，然后初始化 USB A 设备会话。
5. USB 受电外设从 USB 插座移除。一旦 USB Type-C®控制器检测到附设的外设（参见 [7] 获取详细信息）：
  - a. USB Type-C®控制器打开电源开关 (Q1/Q2)，它触发一个中断给 STPMIC1BPQR，激活 VBUS 放电路径，然后进入 Unattached.SRC 状态。
  - b. 一旦 VBUS 放电路径被激活，VBUS 电压变低。
6. 一旦来自 USB Type-C®控制器的中断被软件捕获：
  - a. 软件终止 USB A 设备会话。
  - b. 软件禁用 STPMIC1BPQR PWR\_USB\_SW 电源开关。PWR\_USB\_SW 的内部下拉由 STPMIC1BPQR 自动启用，导致 VBUS\_DR 被拉低：VBUS\_DR 下降（参见第 6.1.3 节）
7. 当 VBUS\_DR 放电时，软件禁用 STPMIC1BPQR 升压转换器：
  - a. BSTOUT 电压下降。
  - b. 一旦 BSTOUT 升压输出电压低于 VBAT，STPMIC1BPQR LDO4 由 VBAT 供电，而 LDO4 输出 (VDD\_USB) 低于 3.3 V（参见第 6.2.3 节）。

**图 14. USB 受电外设供电管理总结**


### 6.2.6 通过 USB Type-C®插座应用实现电池充电和电池监控

本节介绍的是图 12。其中重点论述，当充电器连接到 USB Type-C®插座时，STM32MP15x、STPMIC1BPQR、USB Type-C®控制器，以及电池充电器之间的主要交互。

注意

电池充电算法和电池充电安全不在本应用笔记的讨论范围之内。

图 12 应用原生支持单体电池或并联单元 3.6 V 锂离子或锂聚合物电池。得益于 STPMIC1BPQR BOOST SMPS，也支持所谓的低截止电池。其他电池化学成分也可以支持，但仅限于 STPMIC1BPQR 最低和最高工作电压。

在此应用中，USB 充电器是一个通用 IC。它需要同时支持：

1. 硬件充电：无电电池自主充电。
2. 软件充电：充电和充电监测由 STM32MP15x 软件算法管理。

USB Type-C®控制器 CC1DB 引脚和 CC2DB 引脚分别连接到 CC1 引脚和 CC2 引脚，以启用‘电池耗尽’功能。

如需了解电池接口和存在检测等信息，请参见 第 6.3.1 节

如需了解电池监测信息，请参见 第 6.3.2 节

如需了解关于从‘电量耗尽’模式进行上电和下电，以及电池充电的相关信息，请参见从“电池耗尽”模式进行上电 / 下电，以及电池充电。

#### 电池接口和存在检测

见 第 6.3.1 节

#### 电池监测

见 第 6.3.2 节

从‘电量耗尽’模式进行上电和下电，以及电池充电，面向 USB Type-C®插座应用

见第 6.3.4 节

## 6.3 电池管理

本节详细介绍电池管理方法。它包括：

- 电池存在检测第 6.3.1 节。
- 电池监测（参加第 6.3.2 节）。
- 从电量耗尽的电池进行上电和下电，以及电池充电，面向：
  - 传统 micro-AB (OTG) 或 micro B 插座 (DR)（参见第 6.3.3 节）。
  - Type C 插座（参见第 6.3.4 节）。

### 6.3.1 电池接口和存在检测

该应用本身不支持电池接口，如 Dallas Semiconductor 的 1-wire™或德州仪器的 Single-Wire SDQ™。

电池存在检测和温度监测必须由电池充电器 IC 和/或电池监测 IC（如 STC3115）负责管理。

通常情况下，如果电池不存在，则不允许进行电池预充电。届时，充电器被设为预充电（因为电池不存在，所以  $VBAT = 0$ ，相当于一个电量用尽的电池），且  $VBAT$  电压上升到  $STPMIC1BPQRVINOK\_rise$  阈值以上。因此， $STPMIC1BPQR$  如第 5.2.1 节中描述的那样上电。由于预充电电流通常非常低， $VBAT$  电压在  $STM32MP15x$  启动后立即下降并开始消耗电源，导致系统不稳定并可能崩溃。

### 6.3.2 电池监测

该应用内嵌一个  $STC3115$  电池剩余电量指示 IC，用于准确监测电池的充电状态 (SOC)。

为了监测电池充电过程中的温度， $STC3115$  内嵌一个温度传感器。另外，如果电池组内嵌 NTC 电阻（负温度系数），则可以使用  $STM32MP15xADC$  输入将 NTC 电阻转换为温度。

### 6.3.3 传统 micro-AB (OTG) 或 micro B 插座 (DR) 从“电池耗尽”模式进行上电/下电，以及电池充电

当电池组完全放电， $VBAT = 0$ （电池组内嵌的保护电路打开，使  $VBAT$  处于高阻抗状态）或放电后的电池的  $VBAT < STPMIC1BPQRVINOK\_rise$  阈值时，应用无法打开（参见第 5.2.1 节）。为了保持电池的化学性质，需要以低充电电流进行预充电循环。该循环完全由电池充电器 IC 管理，直至应用能够打开 ( $VBAT > STPMIC1BPQRVINOK\_rise$ )。应用程序进入运行状态后， $STM32MP15x$  软件管理和监控电池充电。该过程详见图 15

当系统在电池已放电且没有充电器的情况下运行时，可以对  $STPMIC1BPQRVINLOW\_rise$  阈值进行编程以发送中断到  $STM32MP15x$ （当  $VBAT$  超过  $STPMIC1BPQRVINLOW\_rise$  阈值）。 $STM32MP15x$  软件必须在电池电压变得非常低之前管理该中断。为了保护电池的化学性质免受过低电压的影响，可以对  $STPMIC1BPQRVINOK\_Fall$  阈值进行编程，以自动关闭应用（参见第 5.2.1 节）。

1. 应用没有电源：电池组存在但已完全放电： $VBAT = 0$ （电池电量耗尽：电池组的内嵌保护电路打开）。
2. 来自 USB 主机外设（例如：来自个人计算机 (PC)）的 USB 标准下行端口或 USB 充电器外设的线缆插入 USB 插座：
  - a.  $VBUS\_DR$  电压上升到 ~5V
  - b. 一旦  $VBUS\_DR$  电压稳定下来，电池充电器以非常低的充电电流开始预充电循环。电池电压 ( $VBAT$ ) 上升
  - c. 一旦电池组内嵌的保护电路检测到  $VBAT$  电压高于电池单元电压，它会禁用将电池单元连接到  $VBAT$  的保护。电池组开始充电，使  $VBAT$  电压缓慢上升。
3. 一旦  $VBAT$  电压高于  $VIN\_POR\_rise$ ：
  - a.  $STPMIC1BPQR$  初始化，确认  $NRST$  有效并预加载其 NVM 内容
  - b. 由于  $STPMIC1BPQR$  在 NVM 中默认设置“ $AUTO\_TURN\_ON$ ”位，因此只要  $VBAT < VINOK\_rise$ ， $STPMIC1BPQR$  直接进入“检查 & 加载”状态（参见[2]获取详细信息）。
4. 一旦  $VBAT$  电压高于  $VINOK\_rise$ ， $STPMIC1BPQR$  执行上电时序，然后释放  $NRST$  信号（参见图 6 获取关于上电时序的详细信息）
5. 一旦  $STM32MP15x$   $NRST$  上升完毕， $STM32MP15x$  软件开始执行并初始化应用程序。

**6. STM32MP15x 软件初始化 USB 充电器 IC:**

- a. 停止预充电
- b. 设置最大充电电流、最高充电电压等快速充电参数。这些设置取决于 USB 供电电流能力、电池容量等。
- c. 当电池放电后 ( $V_{BAT} < \text{最高充电电压设置}$ )，USB 充电器 IC 以恒定电流模式为电池充电。电池电压缓慢上升。

*注意* 充电过程中的电池监测不在本文档的讨论范围之内。

**7. 一旦 VBAT 电压高于最高充电电压设置:**

- a. USB 充电器 IC 切换到恒定电压充电模式。
- b. 在充电过程中，电池电压保持稳定，充电电流减小，直至充电检测结束。

**8. 一旦充电电流低于  $I_{\text{EndOfCharge}}$  阈值，USB 充电器 IC 停止充电，然后进入 NO CHARGE 状态。**
**9. 来自 USB 主机外设或 USB 充电器外设的线缆从 USB 插座上移除:**

- a. VBUS\_DR 电压下降，USB 充电器 IC 进入高阻抗状态。
- b. 应用程序运行，电池正在放电。

**10. 一旦 VBAT 电压低于 STPMIC1BPQR VINLOW\_rise 阈值:**

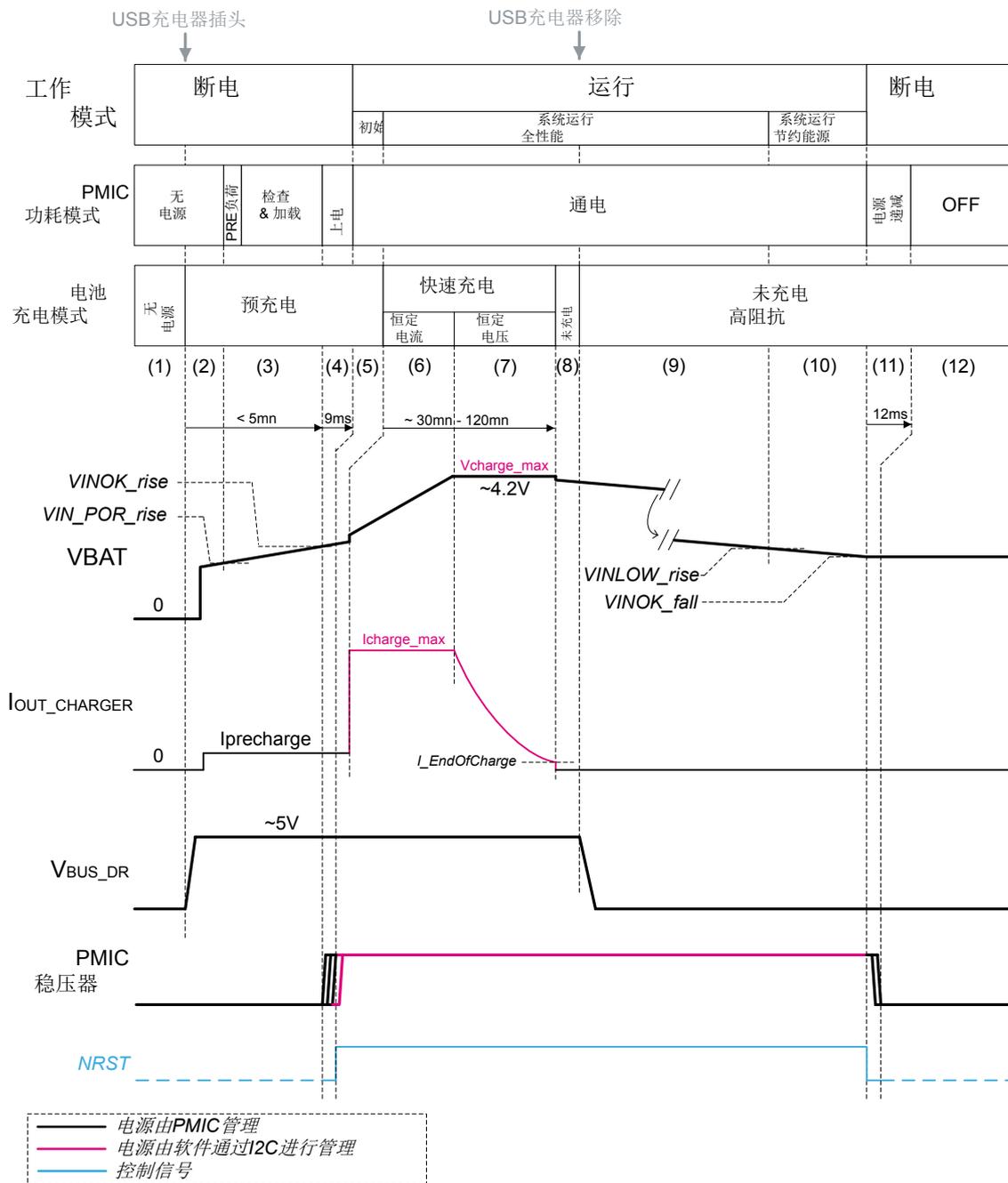
- a. STPMIC1BPQR 触发一个给 STM32MP15x 的中断，警告 STM32MP15x 电池电压为低。
- b. STM32MP15x 软件可能进入节能模式

**11. 一旦 VBAT 电压低于 STPMIC1BPQR VINOK\_fall 阈值:**

- a. 关闭条件出现 STPMIC1BPQR
- b. STPMIC1BPQR 确认 NRST 有效，并执行下电时序（参见图 6 获取关于下电时序的详细信息）

**12. STPMIC1BPQR 处于关闭模式:**

- a. 应用程序处于断电模式。
- b. 只要 VBAT 电压低于 VINOK\_rise，STPMIC1BPQR 不能上电。

**图 15. 传统插座从“电池耗尽”模式上电 / 下电**


### 6.3.4 Type C 插座应用从“电池耗尽”模式进行上电 / 下电，以及电池充电

当电池组完全放电， $VBAT = 0$ （电池组内嵌的保护电路打开，使  $VBAT$  处于高阻抗状态）或电池高度放电， $VBAT < STPMIC1BPQRVINOK\_rise$  阈值时，应用无法打开（参见第 5.1.1 节）。为了保持电池的化学性质，需要以低充电电流进行预充电循环。该循环完全由电池充电器 IC 管理，直至应用能够打开 ( $VBAT > STPMIC1BPQRVINOK\_rise$ )。应用程序进入运行状态后，STM32MP15x 软件管理和监控电池充电。

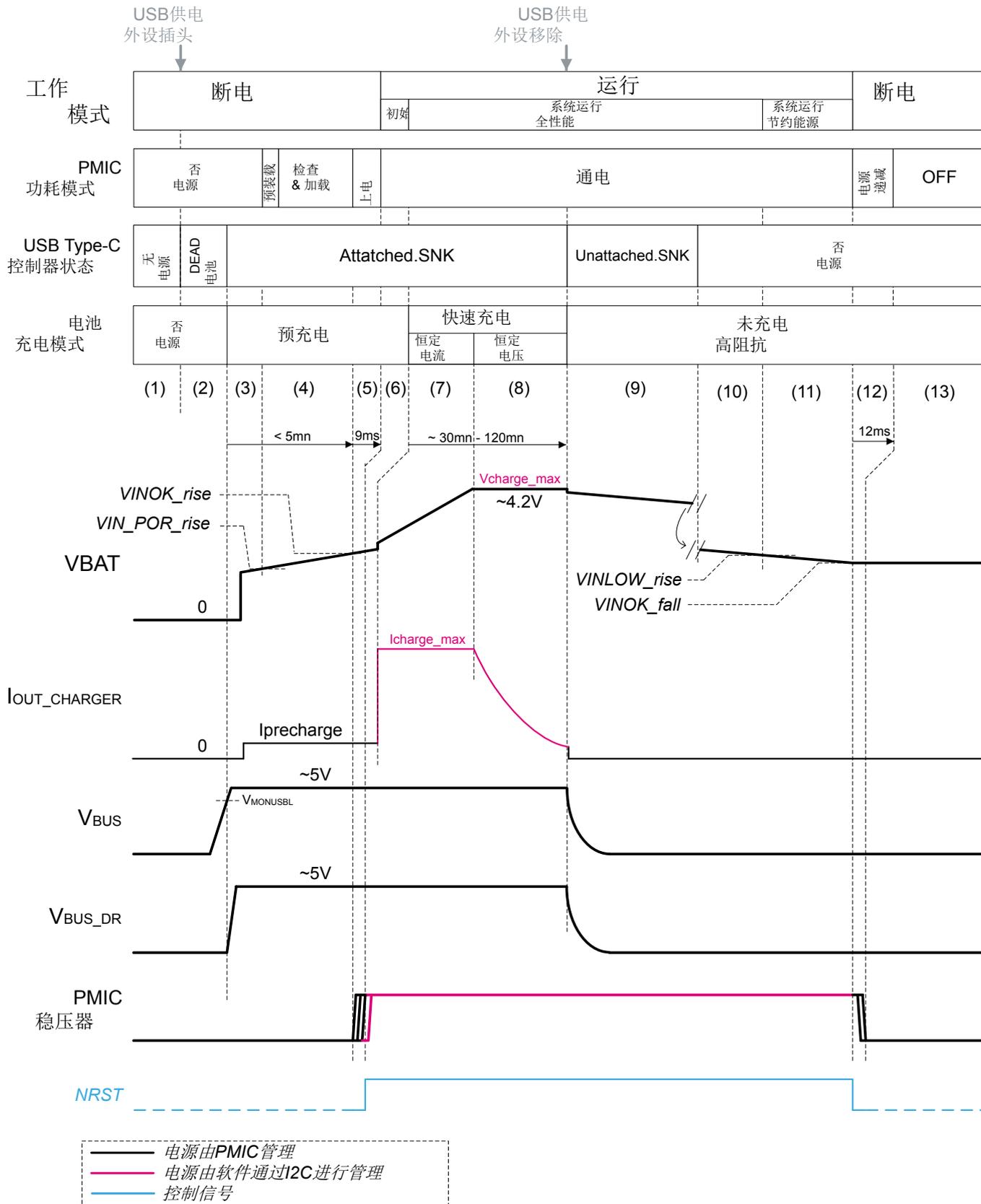
当系统在电池已放电且没有充电器的情况下运行时，可以对  $STPMIC1BPQRVINLOW\_rise$  阈值进行编程以发送中断到 STM32MP15x（当  $VBAT$  超过  $STPMIC1BPQRVINLOW\_rise$  阈值）。STM32MP15x 软件必须管理该中断，直至电池电压变得非常低。为了保护电池的化学性质免受过低电压的影响，可以对  $STPMIC1BPQRVINOK\_Fall$  阈值进行编程，以自动关闭应用（参见第 5.1.1 节）。

1. 应用没有电源：电池组存在但已完全放电： $VBAT = 0$ （电池电量耗尽：电池组的内嵌保护电路打开）。
2. 来自 USB 供电外设的电缆插入 USB 插座：
  - a. USB Type-C®控制器进入电池耗尽模式，其广播为 Unattached.SNK。
  - b. 一旦 USB 供电外设检测到与应用程序的连接，则其为  $VBUS$  供电。  
USB 供电外设可以参照个人电脑 (PC) 或 USB Type-C®充电器的 USB Type-C®端口。
3. 一旦  $VBUS$  高于  $VMONUSBL$  阈值（参见[7]获取详细信息）：
  - a. USB Type-C®控制器进入 Attached.SNK 状态，它闭合电源开关 (Q1/Q2) 并触发一个中断给 STM32MP15x（参见第 6.2.1 节）。
  - b.  $VBUS\_DR$  电压上升到  $\sim 5V$ 。
  - c. 一旦  $VBUS\_DR$  电压稳定下来，电池充电器以非常低的充电电流开始预充电循环。电池电压 ( $VBAT$ ) 上升。
  - d. 一旦电池组内嵌的保护电路检测到  $VBAT$  电压高于电池单元电压，它会禁用将电池单元连接到  $VBAT$  的保护。电池组开始充电，导致  $VBAT$  电压缓慢上升。
4. 一旦  $VBAT$  电压高于  $VIN\_POR\_rise$ ：
  - a.  $STPMIC1BPQR$  初始化，确认  $NRST$  有效并预加载其 NVM 内容。
  - b. 由于  $STPMIC1BPQR$  在 NVM 中默认设置“AUTO\_TURN\_ON”位，因此只要  $VBAT < VINOK\_rise$ ， $STPMIC1BPQR$  直接进入“检查 & 加载”状态（参见[2]获取详细信息）。
5. 一旦  $VBAT$  电压高于  $VINOK\_rise$ ， $STPMIC1BPQR$  执行上电时序，然后释放  $NRST$  信号（参见图 5 获取关于上电时序的详细信息）
6. 设置完 STM32MP15x  $NRST$  之后，STM32MP15x 软件开始执行并初始化应用程序：
7. STM32MP15x 软件初始化 USB 充电器 IC：
  - a. 停止预充电
  - b. 设置最大充电电流、最高充电电压等快速充电参数。具体设置取决于 USB 供电电流能力、电池容量等。
  - c. 当电池放电后 ( $VBAT < \text{最高充电电压设置}$ )，USB 充电器 IC 以恒定电流模式为电池充电。电池电压缓慢上升。

*注意* 充电过程中的电池监测不在本文档的讨论范围之内。

8. 一旦  $VBAT$  电压高于最高充电电压设置：
  - a. USB 充电器 IC 切换到恒定电压充电模式。
  - b. 在充电过程中，电池电压保持稳定，充电电流减小，直至充电检测结束。
9. USB 供电外设的电缆从 USB 插座上移除。一旦 USB Type-C®控制器检测到外设被移除（参见[7]获取详细信息）：
  - a. USB Type-C®控制器断开电源开关 (Q1/Q2)，触发给 STM32MP15x 的中断，激活  $VBUS$  放电路径，进入 Unattached.SNK 状态。
  - b. 一旦  $VBUS$  放电路径被激活， $VBUS$  电压变低。
  - c.  $VBUS\_DR$  电压下降。如果电池正在充电，则 USB 充电器 IC 处于高阻抗状态。
  - d. 应用程序运行，电池正在放电。
10. 一旦  $VBAT$  电压低于 USB Type-C®控制器  $VSYS$  最低电压（参见[7]），USB Type-C®控制器关闭电源。
11. 一旦  $VBAT$  电压低于  $STPMIC1BPQR VINLOW\_rise$  阈值：
  - a.  $STPMIC1BPQR$  触发一个给 STM32MP15x 的中断，警告 STM32MP15x 电池电压为低。
  - b. STM32MP15x 软件可能进入节能模式。
12. 一旦  $VBAT$  电压低于  $STPMIC1BPQR VINOK\_fall$  阈值：
  - a. 关闭条件在  $STPMIC1BPQR$  中出现。
  - b.  $STPMIC1BPQR$  确认  $NRST$  有效，并执行下电时序（参见图 5 获取关于下电时序的详细信息）

13. STPMIC1BPQR 处于关闭模式:
- a. 应用程序处于断电模式。
  - b. 只要 VBAT 电压低于 VINOK\_rise, STPMIC1BPQR 不能上电。

**图 16. Type C USB 插座从“电池耗尽”模式上电 / 下电**


## 修订历史

表 5. 文档修订历史

日期	版本	变更
2020 年 2 月 3 日	1	初始版本。
2021 年 1 月 22 日	2	修改了： <ul style="list-style-type: none"> <li>• 文档标题</li> <li>• 简介</li> <li>• 第 2 节: 概述</li> <li>• 第 2.1 节: 参考文档</li> <li>• 第 4.2.2 节: VDDCORE 电源域 (1.2 V / 1.35 V)</li> <li>• 第 4.2.6 节: VDD_SD 电源域 (2.9 V)</li> <li>• 第 4.2.8 节: VDD_AUDIO (1.8 V)、VDD_LCD (2.8 V) 电源域</li> <li>• 第 4.3 节: 控制信号与接口, 用于连接 STM32MP157 和 STPMIC1BPQR</li> <li>• 第 5.1.1 节: 应用通电 / 断电条件</li> <li>• 第 5.2.1 节: 通过电池插入实现上电</li> <li>• 第 5.3.2 节: 待机模式</li> <li>• 第 5.4.1 节: 故障恢复管理或用户复位顺序</li> <li>• 第 6.1.1 节: USB 外设插头检测和外设移除检测管理</li> </ul>

## 目录

<b>1</b>	<b>概述</b>	<b>2</b>
<b>2</b>	<b>概述</b>	<b>3</b>
<b>2.1</b>	<b>参考文档</b>	<b>3</b>
<b>3</b>	<b>词汇表</b>	<b>4</b>
<b>4</b>	<b>壁式适配器供电电池应用参考设计</b>	<b>5</b>
<b>4.1</b>	<b>电池作为主电源</b>	<b>7</b>
<b>4.2</b>	<b>功率分配</b>	<b>7</b>
<b>4.2.1</b>	<b>VDD 电源域 (1.8 V)</b>	<b>7</b>
<b>4.2.2</b>	<b>VDDCORE 电源域 (1.2 V / 1.35 V)</b>	<b>7</b>
<b>4.2.3</b>	<b>VDD_USB 电源域 (3.3 V)</b>	<b>7</b>
<b>4.2.4</b>	<b>VDD1_DDR (1.8 V)、VDD2_DDR (1.2 V)、VREF_DDR (0.6 V) 电源域</b>	<b>8</b>
<b>4.2.5</b>	<b>VDD_eMMC 电源域 (2.9 V)</b>	<b>10</b>
<b>4.2.6</b>	<b>VDD_SD 电源域 (2.9 V)</b>	<b>10</b>
<b>4.2.7</b>	<b>VBUS_DR 电源域 (5 V)</b>	<b>10</b>
<b>4.2.8</b>	<b>VDD_AUDIO (1.8 V)、VDD_LCD (2.8 V) 电源域</b>	<b>11</b>
<b>4.3</b>	<b>控制信号与接口，用于连接 STM32MP157 和 STPMIC1BPQR</b>	<b>12</b>
<b>4.3.1</b>	<b>采用可移除电池组的应用中的唤醒引脚管理</b>	<b>13</b>
<b>5</b>	<b>电源管理</b>	<b>14</b>
<b>5.1</b>	<b>工作模式</b>	<b>14</b>
<b>5.1.1</b>	<b>应用通电 / 断电条件</b>	<b>14</b>
<b>5.1.2</b>	<b>STPMIC1BPQR restart_request 和 mask_reset 选项</b>	<b>15</b>
<b>5.2</b>	<b>应用上电 / 下电时序</b>	<b>16</b>
<b>5.2.1</b>	<b>通过电池插入实现上电</b>	<b>16</b>
<b>5.2.2</b>	<b>从 STPMIC1BPQR 关闭模式上电</b>	<b>18</b>
<b>5.2.3</b>	<b>电池移除导致下电</b>	<b>18</b>
<b>5.3</b>	<b>低功耗 模式管理</b>	<b>19</b>
<b>5.3.1</b>	<b>LPLV-stop 模式</b>	<b>19</b>
<b>5.3.2</b>	<b>待机模式</b>	<b>21</b>
<b>5.4</b>	<b>用户复位和崩溃恢复管理</b>	<b>23</b>
<b>5.4.1</b>	<b>故障恢复管理或用户复位顺序</b>	<b>23</b>
<b>5.5</b>	<b>软件管理示例</b>	<b>25</b>
<b>6</b>	<b>USB 端口管理</b>	<b>26</b>
<b>6.1</b>	<b>USB 端口使用传统 micro-AB (OTG) 或 micro-B 插座 (DR)</b>	<b>26</b>
<b>6.1.1</b>	<b>USB 外设插头检测和 外设移除检测管理</b>	<b>26</b>
<b>6.1.2</b>	<b>在 USB 主机模式下从电池电压生成 VBUS_DR 电压</b>	<b>27</b>

6.1.3	VBUS_DR 放电下拉功能管理 STPMIC1BPQR .....	28
6.1.4	STM32MP15x USB PHY 由放电后的电池（低于 3.3 V）供电 .....	28
6.1.5	USB 主机外设电源管理总结 .....	28
6.1.6	USB 设备外设电源管理总结 .....	30
6.1.7	通过 USB micro-B 插座应用实现电池充电和电池监控 .....	31
6.2	USB 端口（使用 USB Type-C®插座） .....	31
6.2.1	USB 外设插头检测和移除检测管理 .....	33
6.2.2	在供电模式下从电池电压生成 VBUS 电压 .....	34
6.2.3	STM32MP15x USB PHY 由放电后电压低于 3.3 V 的电池供电 .....	34
6.2.4	USB 供电外设电源管理总结 .....	34
6.2.5	USB 受电外设供电管理总结 .....	36
6.2.6	通过 USB Type-C®插座应用实现电池充电和电池监控 .....	37
6.3	电池管理 .....	38
6.3.1	电池接口和存在检测 .....	38
6.3.2	电池监测 .....	38
6.3.3	传统 micro-AB (OTG) 或 micro B 插座 (DR) 从“电池耗尽”模式进行上电/下电，以及电 池充电 .....	38
6.3.4	Type C 插座应用从“电池耗尽”模式进行上电 / 下电，以及电池充电 .....	41
修订历史 .....		44

## 图一览

图 1.	STM32MP15x 和 STPMIC1BPQR (带 lpDDR2/3、eMMC、SD-card 和 USB DR) STM32MP15x 以及 STPMIC1BPQR (带 DDR3L、启动 flash、SD-card 和 2x USB HS) . . . . .	6
图 2.	USB 设备模式下的 VBUS_DR 电源路径. . . . .	11
图 3.	USB 主机模式下的 VBUS_DRD 电源路径. . . . .	11
图 4.	可移除电池组应用的唤醒引脚控制电路. . . . .	13
图 5.	上电/下电序列. . . . .	17
图 6.	LPLV-Stop 模式顺序 . . . . .	20
图 7.	待机模式顺序 . . . . .	22
图 8.	故障恢复顺序 . . . . .	24
图 9.	使用 USB micro-AB 或 micro-B 插座实现 USB 端口管理 . . . . .	26
图 10.	USB 主机外设电源管理总结 . . . . .	29
图 11.	USB 设备外设电源管理总结 . . . . .	31
图 12.	使用 USB Type-C®插座进行 USB 端口管理 . . . . .	32
图 13.	USB 供电外设电源管理总结 . . . . .	35
图 14.	USB 受电外设供电管理总结 . . . . .	37
图 15.	传统插座从“电池耗尽”模式上电 / 下电 . . . . .	40
图 16.	Type C USB 插座从“电池耗尽”模式上电 / 下电 . . . . .	43

## 表一览

表 1.	参考文档.....	3
表 2.	词汇表.....	4
表 3.	应用程序工作模式.....	14
表 4.	STPMIC1BPQR 电源管理选项.....	25
表 5.	文档修订历史.....	44

**重要通知 - 请仔细阅读**

意法半导体公司及其子公司（“ST”）保留随时对 ST 产品和/或本文档进行变更、更正、增强、修改和改进的权利，恕不另行通知。买方在订货之前应获取关于意法半导体产品的最新信息。意法半导体产品的销售依照订单确认时的相关意法半导体销售条款。

买方自行负责对意法半导体产品的选择和使用，意法半导体概不承担与应用协助或买方产品设计相关的任何责任。

意法半导体不对任何知识产权进行任何明示或默示的授权或许可。

转售的意法半导体产品如有不同于此处提供的信息的规定，将导致意法半导体针对该产品授予的任何保证失效。

ST 和 ST 标志是意法半导体的商标。关于意法半导体商标的其他信息，请访问 [www.st.com/trademarks](http://www.st.com/trademarks)。其他所有产品或服务名称是其各自所有者的财产。

本文档中的信息取代本文档所有早期版本中提供的信息。

© 2021 STMicroelectronics - 保留所有权利