Application Note

如何使用 TI 处理器设计稳健的 OTA 系统

Zekun Bai

摘要

空中 (OTA) 固件更新是现代嵌入式系统的一项关键功能，使器件能够在部署后接收软件更新。但是，OTA 过程中出现的任何问题都可能导致器件无法使用。

TI 处理器系列（如 AM62 和其他 Sitara™ 处理器）提供了强大的多核架构和丰富的外设接口，因此特别适合需要高度可靠 OTA 更新的工业和汽车应用。这些处理器集成了 ARM® Cortex® R5/M4 内核和 A53/A72 内核，支持各种存储器接口和引导选项，为稳健的 OTA 系统提供了硬件基础。

本应用手册介绍了如何使用 TI 处理器设计更稳健、更灵活的 OTA 系统，从而避免常见的 OTA 故障场景。

内容

1 传统 OTA 流程和分析 2
1.1 典型的 OTA 故障场景 2
1.2 传统 OTA 过程的局限性 2
2 TI 处理器 OTA 系统的创新设计 3
2.1 双插槽设计增强了稳健性 3
2.2 状态标志系统 3
2.3 回滚机制 3
2.4 关键区域保护 4
3 改进的 OTA 过程 5
4 总结 6
5 参考资料 7

商标

1 传统 OTA 流程和分析

1.1 典型的 OTA 故障场景

在实际应用中，OTA 更新可能导致片上系统 (SoC) 变砖。通过分析，可以发现以下常见问题：

- 引导介质上的电源轨不稳定，导致应用程序损坏。
- 缺少备份机制；一旦原始应用程序损坏，就无法恢复。
- 没有状态标志来指示应用程序状态和版本信息。
- 缺乏监测机制来检查 OTA 后的首次引导。

1.2 传统 OTA 过程的局限性

传统的 OTA (Over-The-Air) 过程通常涉及 ROM 引导、二级引导加载程序 (SBL) 和应用程序二进制文件。更新时，新应用程序会直接覆盖现有应用程序，这种方式缺乏稳健性。

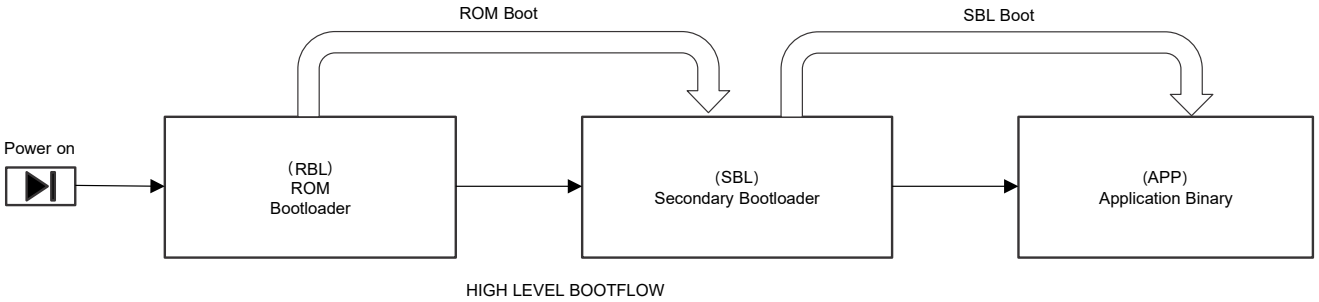

图 1-1. 传统上电启动过程

传统的 OTA (Over-The-Air) 更新需要在上电时连接新的引导介质，例如 SD 卡。SD 卡存储要升级的应用程序和引导加载程序 (SBL)。ROM 从 SD 卡加载文件，覆盖原始引导介质上的旧应用程序位置。

此过程的主要问题包括：

- 无备份机制：如果新应用程序损坏，则无法回滚。
- 无状态标志：无法跟踪应用程序状态和版本。
- 无监测机制：无法监测 OTA 后的首次引导。

2 TI 处理器 OTA 系统的创新设计

2.1 双插槽设计增强了稳健性

TI 处理器支持双插槽设计：

- 插槽 A 存储原始应用程序，设置为只读以提高稳健性。
- 插槽 B 用于 OTA 请求并存储新应用程序。
- 该设计可确保即使新应用程序损坏，系统仍然可以恢复到原始应用程序。

2.2 状态标志系统

引入了标志机制来指示 OTA 状态：

- 标志 = 0：SBL 从插槽 A 加载应用程序。
- 标志 = 1：SBL 从插槽 B 加载新应用程序。

客户还可以定义更多标志，例如表示 OTA 启动、OTA 进行中和 OTA 完成的标志。这使得系统的 OTA 状态对外界更加清晰，防止 OTA 过程变成黑盒。

2.3 回滚机制

在 AB 分区机制的基础上，向芯片引导内核添加了由 WDG 触发的代码回滚逻辑。TI 处理器支持的回滚机制主要通过以下关键要素来实现。

1.看门狗计时器监测：

- SBL 在加载新应用程序时设置一个看门狗计时器。
- 该计时器用作安全措施；如果新应用程序无法正常运行，则会触发系统复位。

2.确认信号机制：

- 新应用程序成功启动后，必须向 R5F-0 发送确认信号 (ACK)。
- 该确认信号表示应用程序已初始化且运行正常。
- 收到确认信号后，R5F-0 清除看门狗计时器，从而完成更新过程。

3.标志状态管理：

- 系统使用持久标志来指示当前应用程序加载位置。
- 标志 = 0：从插槽 A 加载原始应用程序。
- 标志 = 1：从插槽 B 加载新应用程序。
- 在回滚期间，系统将标志复位为 0。

4.自动回滚过程：

- 如果新应用程序未在预定时间内发送确认信号：
- 看门狗计时器到期，触发系统复位。
- 系统将标志重新设置为 0。
- 重新启动后，SBL 检测到标志 = 0 并从插槽 A 加载原始应用程序。

2.4 关键区域保护

ARM 存储器保护单元 (MPU) 通过配置存储器区域的访问权限，将包含引导加载程序的闪存设置为只读。这是设计稳健 OTA 系统时的关键安全措施。

MPU 允许处理器定义存储器区域的属性，包括读取/写入/执行权限。对于存储引导加载程序的闪存区域，可以将其配置为只读，从而防止其他程序（尤其是应用程序）在系统运行期间修改引导加载程序代码。

这种保护机制在 TI 处理器的 OTA 系统设计中尤为重要。分析表明，OTA 故障的一个常见原因是新应用程序损坏了存储器，特别是 SBL（二级引导加载程序）区域的损坏，导致 SoC *变砖*。

通过将 NOR 闪存中的 SBL 引导加载程序区域设置为只读，可有效地防止应用程序意外或恶意修改此关键代码。即使应用程序出现问题，系统仍可以使用正常运行的引导加载程序进行恢复。

这是设计稳健的 OTA 系统、形成完整的保护系统以及双插槽设计、状态标志系统和回滚机制的重要步骤之一。

改进的 NOR 闪存布局包括：双备份 SBL（引导加载程序）和双备份应用程序（业务文件）。

表 2-1. 改进的 NOR 闪存布局

Nor 闪存	文件
0xA	SBL 引导加载程序
0xB	SBL 引导加载程序
插槽 A	应用程序 A
插槽 B	应用程序 B

为了防止存储器损坏导致系统*变砖*，TI 建议为 MPU 中的关键区域（例如 SBL）设置只读属性。

3 改进的 OTA 过程

使用 TI 处理器的稳健 OTA 过程包括：

1. 上电时，SBL 检查标志（最初为 0）并从插槽 A 加载应用程序。
2. 运行应用程序并检查是否存在 OTA 请求。
3. 如果收到请求，则将新应用程序载入插槽 B。
4. 将标志设置为 1 并触发复位。
5. SBL 检查标志（现在为 1）并从插槽 B 加载新应用程序。
6. 新应用程序运行；如果成功运行，该应用程序向 R5F-0 发送确认信号。R5F 清除看门狗计时器，完成更新。
7. 如果没有收到 ACK 确认信号，系统将回滚（标志设置为 0）并从插槽 A 加载原始应用程序。

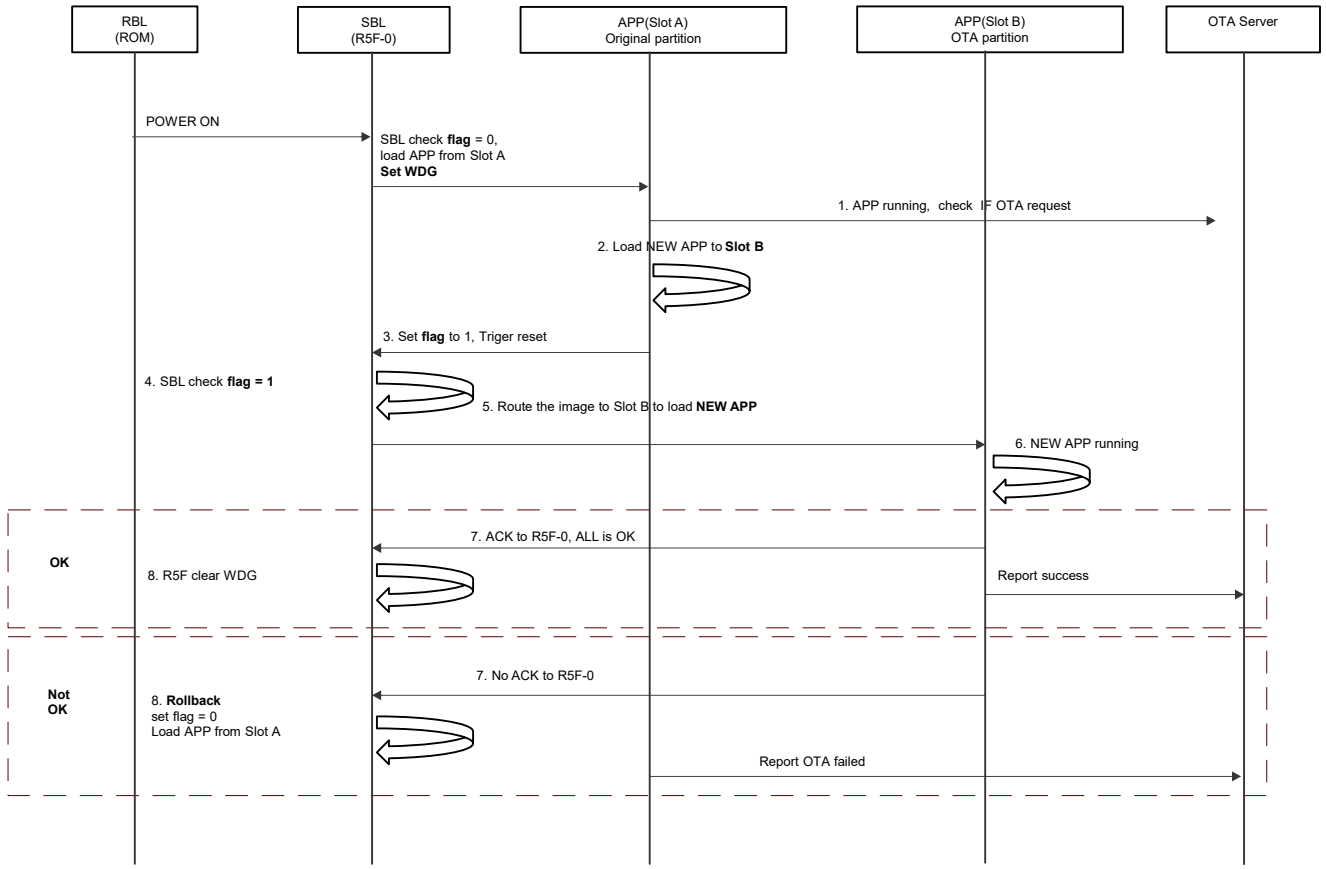

图 3-1. 改进的 OTA 流程

4 总结

传统的 OTA (Over-The-Air) 更新采用了简单的机制，新应用程序会直接覆盖现有应用程序。这种方法缺乏备份机制、状态标志、监测以及对关键区域的保护。如果更新失败，器件可能会变得无法使用，需要人工干预。

本文提出了一种采用双插槽设计的新颖 OTA 过程，其中包含状态标志和自动回滚机制，以验证更新失败时系统是否自动恢复。关键创新包括：双插槽设计、状态标志系统、自动回滚机制和关键区域保护。

该设计的综合优势包括：增强系统稳健性、提高可维护性、无需人工干预、防止连续故障以及保护关键系统元件。

本文提出的 OTA 系统设计方法不仅适用于 AM62 处理器，还可扩展到 TI 的 Jacinto™ 处理器系列（用于汽车信息娱乐系统和 ADAS 应用）、Sitara 处理器系列（用于工业自动化和边缘计算器件）以及其他基于 ARM 的 TI 处理器。

通过这种灵活、稳健的 OTA 系统设计，采用 TI 处理器的器件可实现更可靠的远程更新能力，显著降低现场维护成本并改善最终用户体验。这种设计方法为需要高可靠性 OTA 功能的各种嵌入式系统提供了有价值的参考。

5 参考资料

- 德州仪器 (TI)，AM62x Sitara™ 处理器，数据表。

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、与某特定用途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安全、安保法规或其他要求。

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。对于因您对这些资源的使用而对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，您将全额赔偿，TI 对此概不负责。

TI 提供的产品受 TI 销售条款)、TI 通用质量指南 或 ti.com 上其他适用条款或 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。 除非德州仪器 (TI) 明确将某产品指定为定制产品或客户特定产品，否则其产品均为按确定价格收入目录的标准通用器件。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

最后更新日期：2025 年 10 月