Application Note

AWR294x/AWR2544 主要引导加载程序和辅助引导加载程序

摘要

本文档介绍了 AWR294x/AWR2544 主要引导加载程序和辅助引导加载程序（RBL 和 SBL）的基本详细信息。本文档还介绍了辅助引导加载程序软件的设计和实施注意事项。本文档仅重点关注非安全 AWR294x/AWR2544 器件型号。

内容

商标

1 定义、缩写、首字母缩略词

术语	定义
RBL	ROM 引导加载程序
SBL	辅助引导加载程序
TCM	紧耦合存储器
CAN	控制器局域网
MPU	内存保护单元
AWR294x	AWR2943 和 AWR2944
MSS	主子系统
DSS	DSP 子系统
RSS/BSS	雷达子系统/BIST 子系统
sFLASH/SDF	串行闪存
CCS	Code Composer Studio

2 简介

AWR294x/AWR2544 器件可大致分为如下三个子系统（请参阅图 2-1，图 2-2）：

- 主要子系统 (MSS)：ARM® Cortex®-R5F 和相关外设，托管用户应用程序
- DSP 子系统 (DSS)：TI C66x 和相关外设，托管用户应用程序。DSP 内核 TI C66x 不适用于 AWR2544，因为设计中不包含内核。
- 雷达/BIST 子系统：使用 TI 指定的预定义消息事务进行编程（参考驱动程序：TI 提供的 mmWaveLink）。此子系统为黑盒，不适用于用户应用程序。

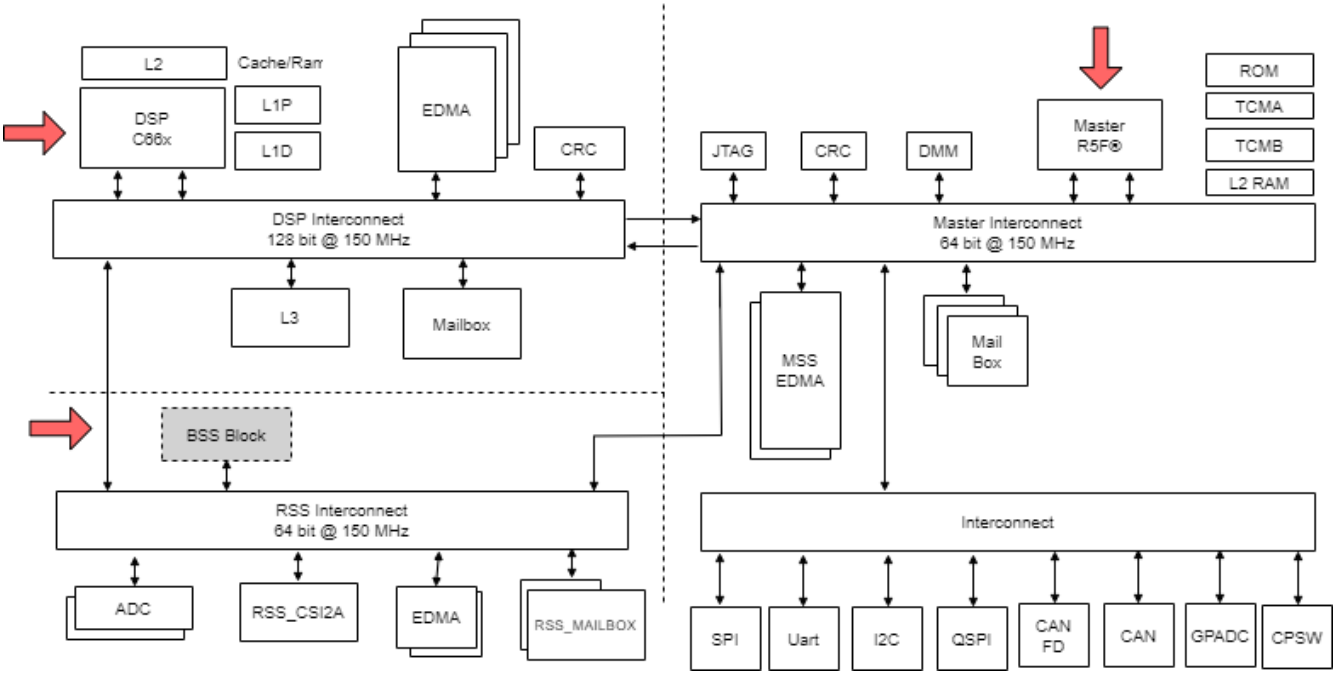

图 2-1. AWR294x 子系统

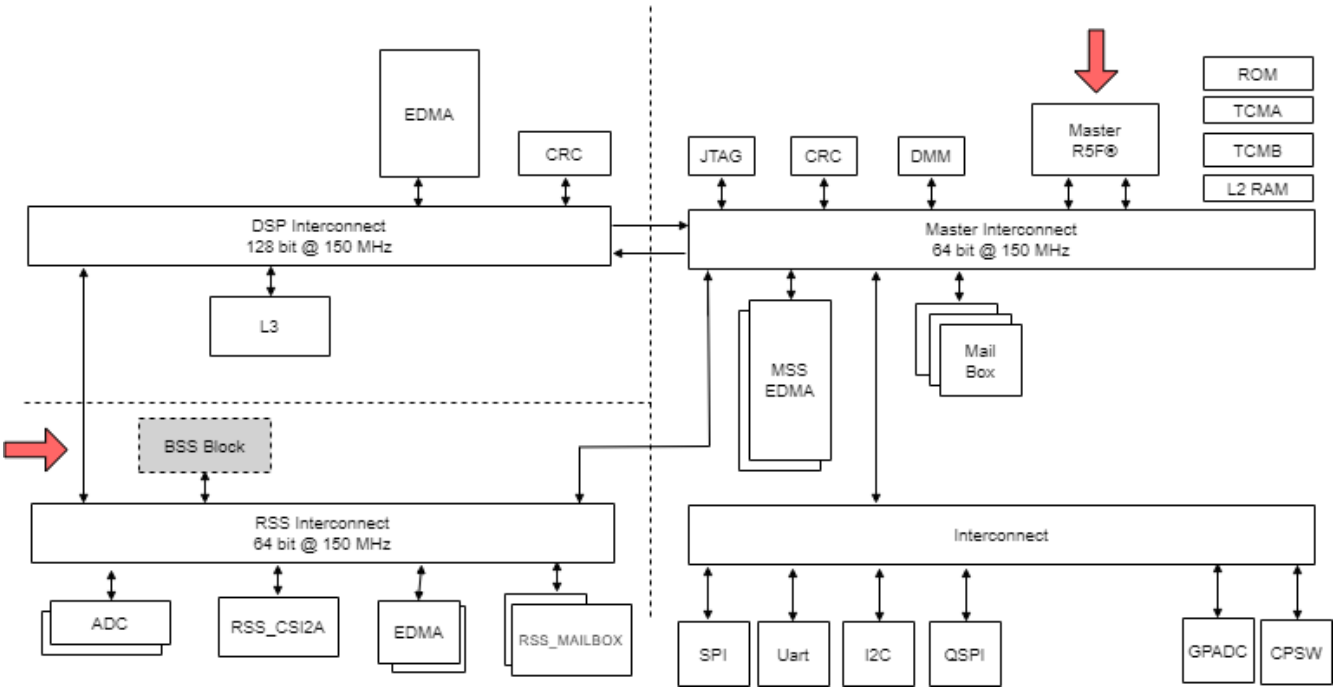

图 2-2. AWR2544 子系统

用户应用程序组件（R5F 和 DSP）应存储在通过四路串行外设接口 (QSPI) 接口连接到 AWR294x 器件的串行数据闪存 (SDF) 中。对于 AWR2544，用户应用元件仅为 R5F 应用图像，因为 DSP 不适用。

主要子系统是在 AWR294x/AWR2544 器件复位取消置位后激活的第一个可编程块。AWR294x/AWR2544 器件的引导加载程序托管在主要子系统的只读存储器 (ROM) 中，并立即取得控制权。

从此时开始，AWR294x/AWR2544 引导加载程序可在两种模式下运行：刷写模式和功能模式。该引导加载程序会检查电源检测 (SOP) I/O（从外部驱动并用于选择特定模式的 SOP 线路）的状态（请参阅表 2-1）。

表 2-1. SOP 线路和引导模式

SOP2 (T17)	SOP1 (R14)	SOP0 (R14)	引导加载程序模式和运行方式
0	0	1	功能或 QSPI 引导模式 RBI 将 SBL 从 QSPI 串行闪存加载到内部 RAM (MSS L2) 并切换控制权。
1	0	1	刷写或 UART 引导模式 RBL 循环运行，允许用户通过 UART(XMODEM) 将 SBL 或用户应用程序直接加载到 RAM 中。此模式可用于将闪存编程器应用程序加载到 RAM 中，然后由 RAM 负责将 SBL 和用户应用程序加载到 SFLASH 中。

RBL 的 UART 引导模式允许外部实体将客户应用程序（闪存编程器）映像加载到 RAM（仅限 MSS L2）中，之后 RAM 会将辅助引导加载程序 (SBL) 和/或用户应用程序下载到 SDF（请参阅图 2-3）。

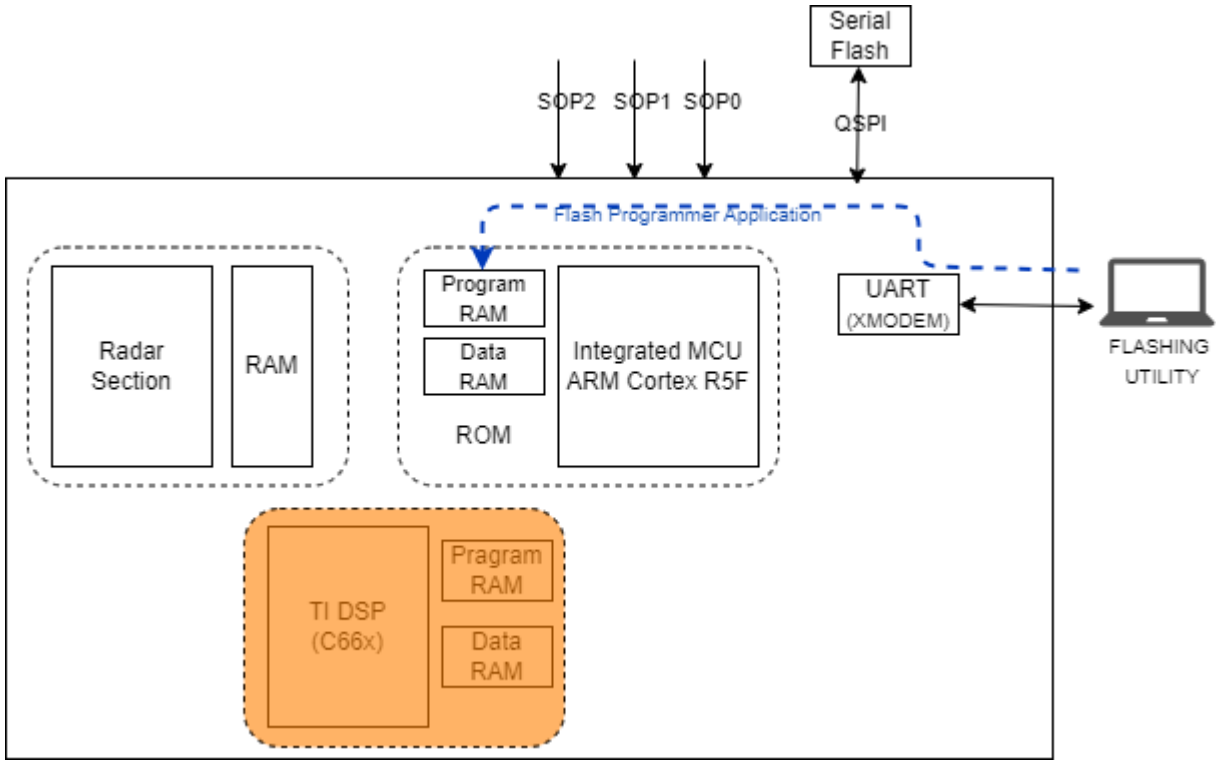

图 2-3. RBL 的刷写模式

RBL 的功能模式会将存储在 SDF 中的映像重定位到 R5F 存储器子系统，此 R5F 映像是 SBL。在此过程结束时，引导加载程序会将控制权交给 R5F 用户定义的 SBL。DSP 和 R4F (BSS) 映像/内核的加载和终止输送（开始执行）由 SBL 负责（请参阅图 2-4）。

备注

请注意，TI C66x 不适用于 AWR2544。

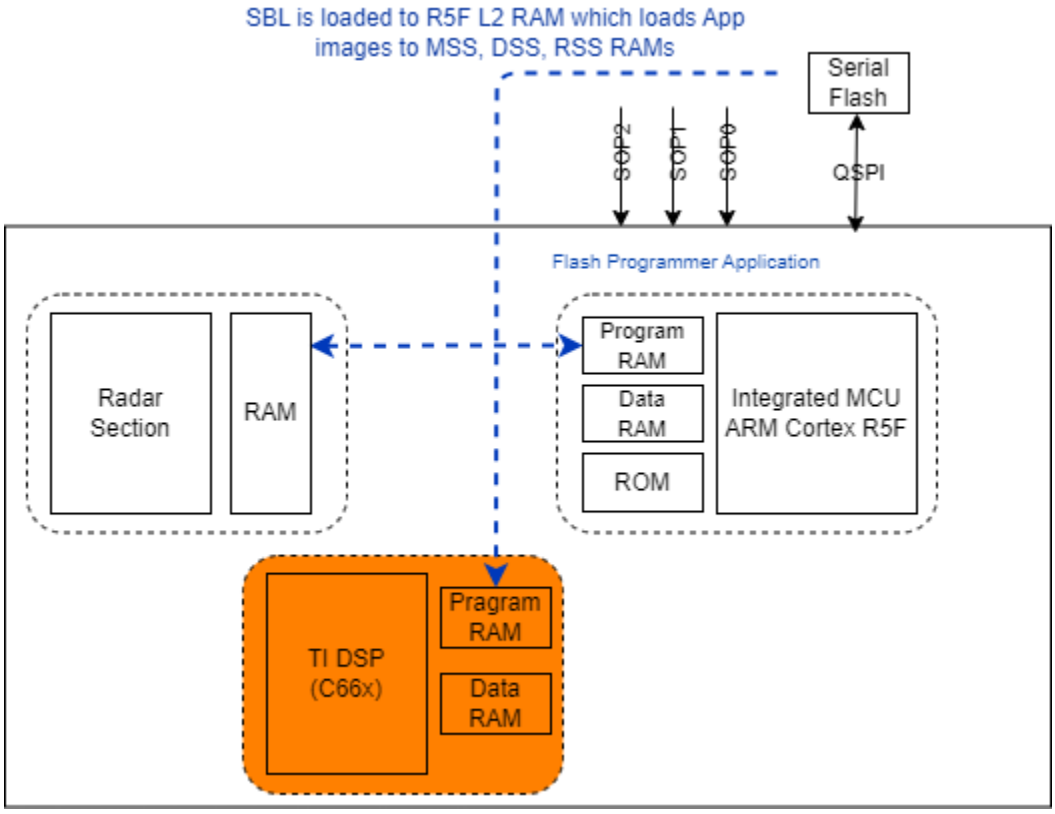

图 2-4. RBL 的功能模式

备注

请注意，TI C66x 不适用于 AWR2544。

要点

- AWR294x 的 RBL 只能加载一个主要用户映像（只能包含 R5F L2 的内容）。
- 此主要用户映像是 SBL，负责向 SDF 加载 MSS、DSS 和 BSS 映像/补丁以及从 SDF 进行相应下载。客户必须致力于通过 SBL 处理多个映像（出厂编程、备份等等）。

3 基本引导加载程序流程

3.1 引导流程简介

在 SOC 上引导用户定义的应用程序涉及下列多个步骤：

- 首先，需要执行多个步骤，将使用编译器+链接器工具链创建的用户应用程序转换为旨在由 SOC 引导的二进制格式。
- 接下来，我们需要将该二进制文件刷写到板载串行闪存中。
- 最后，当 SOC 上电时，执行先前刷写的二进制内容。
- 在功能模式下为器件上电后，引导流程主要分两步进行：
  - ROM 引导：在此步中，RBL 引导从 sFlash 读取的 SBL。
  - SBL 引导：在此步中，辅助引导加载程序会引导从 sFlash 读取的应用程序。
- 请注意，系统应用程序 (即 metaimage) 本身可以包含多个特定于 CPU 的应用程序二进制文件，所有这些二进制文件将协同工作以实现整个系统目标。

3.2 准备引导应用程序

下面显示了将编译器+链接器生成的应用程序 .out 转换为专用于刷写和引导的格式的不同步骤。

- 对于每个 CPU，使用编译器+链接器工具链来创建可通过 CCS/JTAG IDE 加载和运行的应用程序 .out“ELF”文件。
- 然后，使用以下“编译后处理”步骤将应用程序 .out 转换为适用于“闪存”的格式：
  - 对于每个 CPU，out2rpc 会将应用程序可执行文件 (.out) 转换为自定义 TI RPRC (.rprc) 映像。此工具从可执行文件 (*.out) 中剥离已初始化的段，并将这些段置于 SBL 可以理解的紧凑格式中。输出的 RPRC 文件通常比原始可执行文件 (*.out) 小得多。
  - 随后使用 multiCoreGen 将每个 CPU 的所有 RPRC 文件合并为单个 .appimage 文件，该文件是各个 CPU 特有 RPRC 文件的串联结果。
- 然后可以将该 .appimage 刷写到器件中。

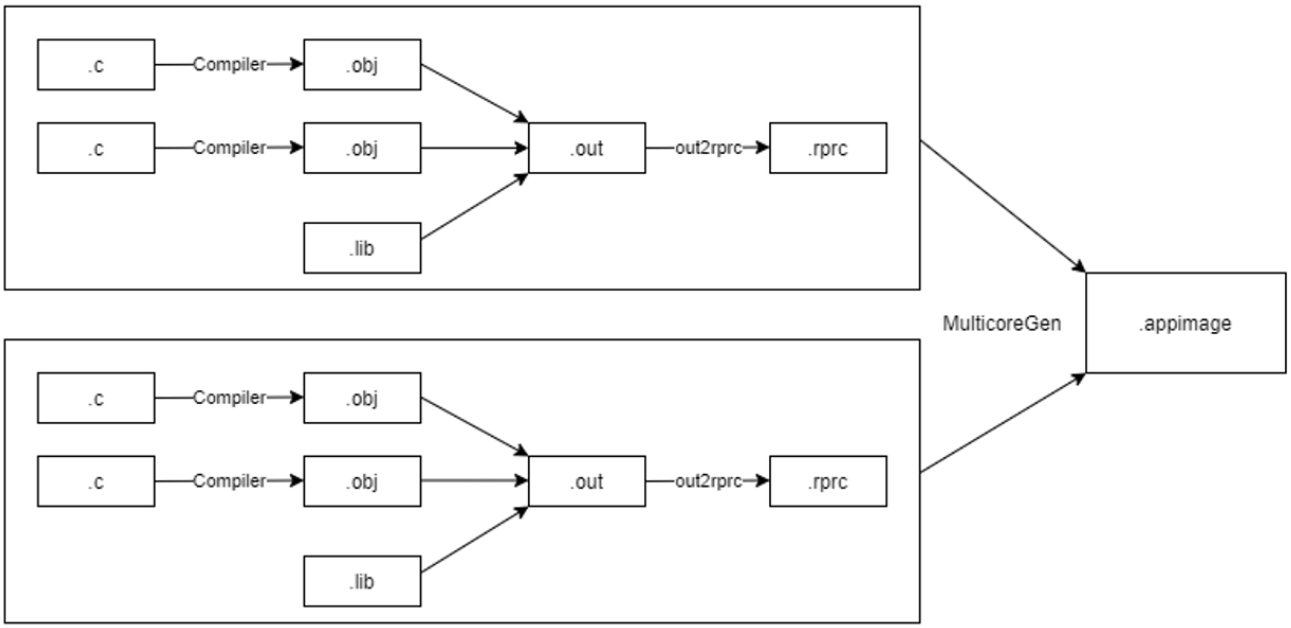

图 3-1. 编译后处理步骤

3.2.1 SBL 映像格式

TI 在 MCU Plus SDK 包中提供了示例 R5F SBL。提供的 SBL 格式为自定义 TI 映像 (.TIIMAGE)。编译器+链接器工具链的输出文件 (.out) 转换为二进制格式 (.bin)，并会在此映像上附加一个证书以生成最终的自定义 TI 映像。

3.2.2 脚本签名

RBL 始终需要签名的引导映像（主要是 SBL）。如果未签名，RBL 将无法引导 SBL。RBL 在刷写或功能模式下加载的任何映像都需要经过签名。创建证书并将证书附加到二进制文件的过程称为对脚本进行签名。

要点

- 在任何工作模式下，RBL 始终需要签名的映像。因此，RBL 加载的任何映像都必须经过签名。在功能模式下，这意味着必须对 sFLASH 中的 SBL 进行签名。在闪存模式下，这意味着由 RBL 加载到 RAM 的闪存编程器映像也必须经过签名。
- 上述格式是 TI 提供的自定义参考格式。由于 SBL 是用户实现的并会引导 .appimage，因此用户可以根据自己的偏好定义 .appimage 的任何格式，并以所需的方式实现 SBL 来解析此映像。

如需了解更多详细信息，请参阅毫米波 MCU Plus SDK 版本的自述文件中的“引导工具”部分。

在刷写 SBL 和应用程序映像后，SOC 通电后的概要流程如下所示。

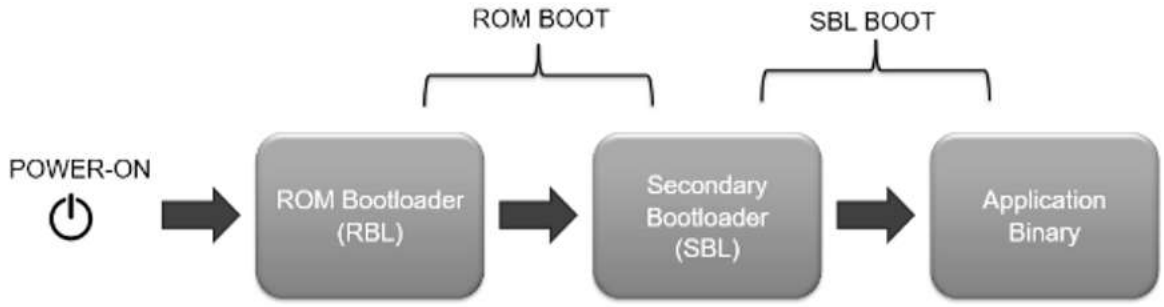

图 3-2. 启动流程概要

3.3 ROM 引导

引导过程包含两个连续步骤：主引导加载程序或 ROM 引导加载程序 (RBL) 过程，然后是辅助引导加载程序 (SBL) 过程。EVM 通电后，ROM 引导加载程序或 RBL 便会开始运行。RBL 是主引导加载程序。RBL 的目标是加载、验证、解密（可选）和启动正版 R5F 软件映像，以实现安全启动目标（在安全型号中）。RBL 过程由 R5F 和 HSM ROM 共同实现，如下图（图 4）所示。RBL 始终需要签名的映像（本例中为 SBL）。

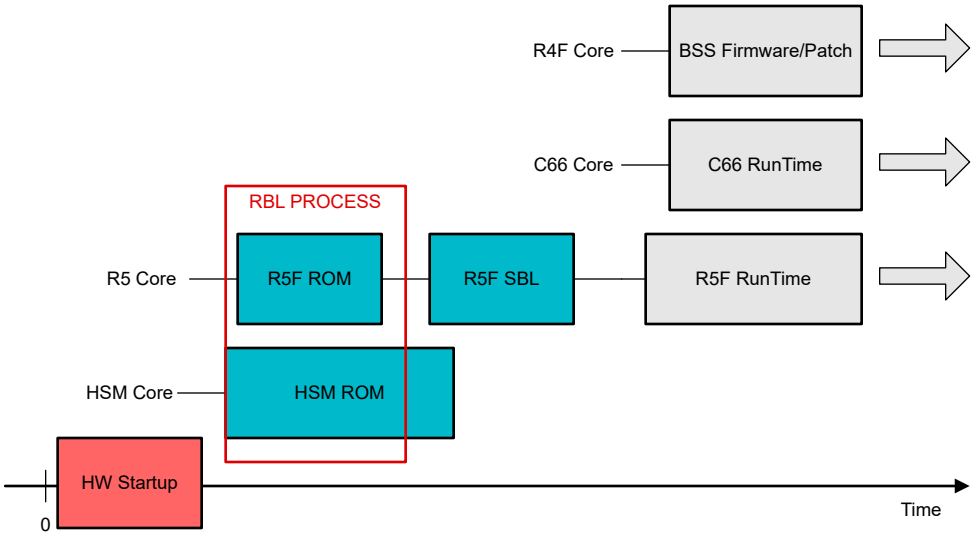

图 3-3. 引导过程

备注

请注意，C66 运行时间和 C66 核心不适用于 AWR2544。

引导流程大致如下所述。

- HSM ROM 是系统复位时首先执行的代码。HSM ROM 在器件初始化期间执行一组针对数据 SRAM、程序 SRAM 和 ROM 代码完整性的自检，并配置 APLL。HSM ROM 还会从复位中释放 R5F。
- 在 HSM ROM 中会对安全 RAM 和公共 RAM 执行 PBIST，并为 HSM 子系统执行 ROM 代码完整性检查。
- 在 R5F RBL 中会对 MSS TCMA、MSS TCMB 和 MSS_L2 存储器执行 PBIST。
- R5F 会检查 SOP 设置并基于持续执行模式。
- 在 UART 引导模式/刷写模式下，RBL 应从 UART 获取闪存编程器（或任何其他相关映像）。闪存编程器通常用于下载 SBL 并将其刷写到 QSPI 闪存中。
- 在 QSPI 引导模式/功能模式下，RBL 将 SBL 从闪存加载到内部 RAM 中并开始执行。

备注

请注意，对于 AWR2544 器件，RBL 不会在 MSS TCM 存储器和 MSS L2 上执行 PBIST。要求用户在 SBL 中执行操作。请检查此实现的 AWR2544 SDK。

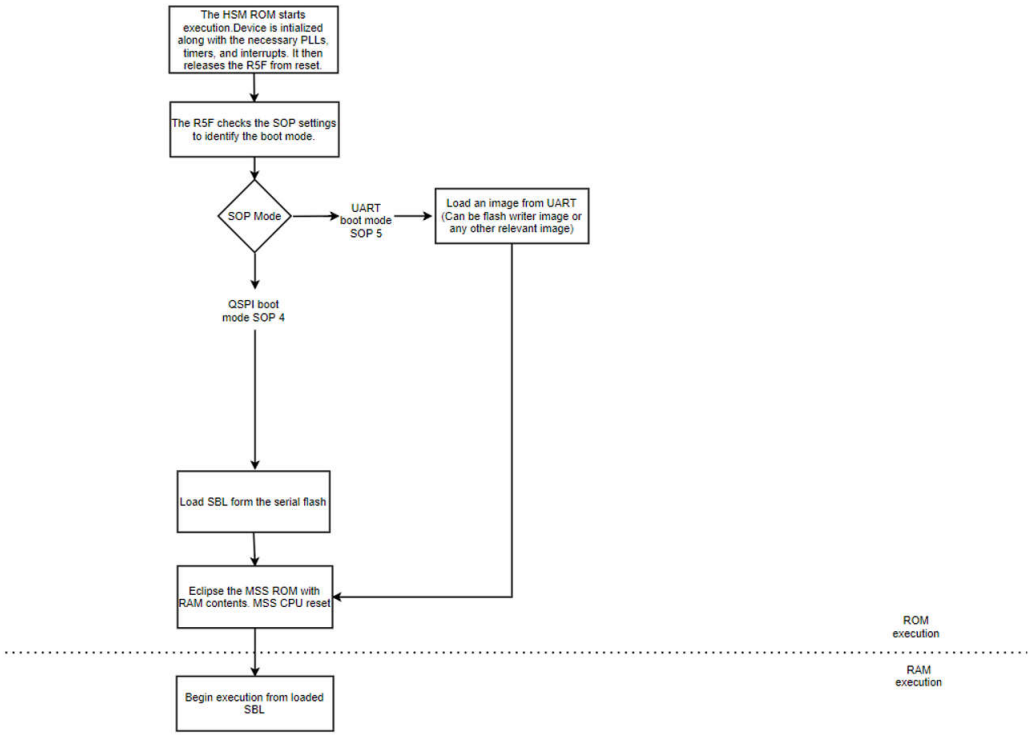

图 3-4. ROM 引导流程

要点

- ROM 引导加载程序仅加载一个映像，并且仅加载到 R5F L2 RAM。
- ROM 引导加载程序通过启动 APLL 来设置根时钟。根时钟的频率为 200MHz

3.3.1 引导模式 - SFLASH

3.3.1.1 映像加载序列

在功能模式下，引导加载程序尝试的第一种引导模式是从 SDF 对映像进行引导加载。此引导模式涉及以下步骤：

1. 对 AWR294x 器件的 QSPI 引脚进行引脚多路复用：
   - QSPI[0]：焊球 U11
   - QSPI[1]：焊球 V11
   - QSPI[2]：焊球 T11
   - QSPI_CLK：焊球 R10
   - QSPI_CS：焊球 U12
2. QSPI 设置为在 (系统时钟 / 5) = (200/5) = 40MHz 下运行。
3. 发出 sFLASH 可发现参数 (SFDP) 命令以检索符合 JEDEC 标准的响应，其中包含有关 sFLASH 功能和命令集的信息。当接收到 SFDP 响应时，该信息用于与 SDF 进行通信并进一步解释内容和加载映像。更多有关 AWR294x/AWR2544 器件支持的闪存型号的信息，请参阅应用手册。

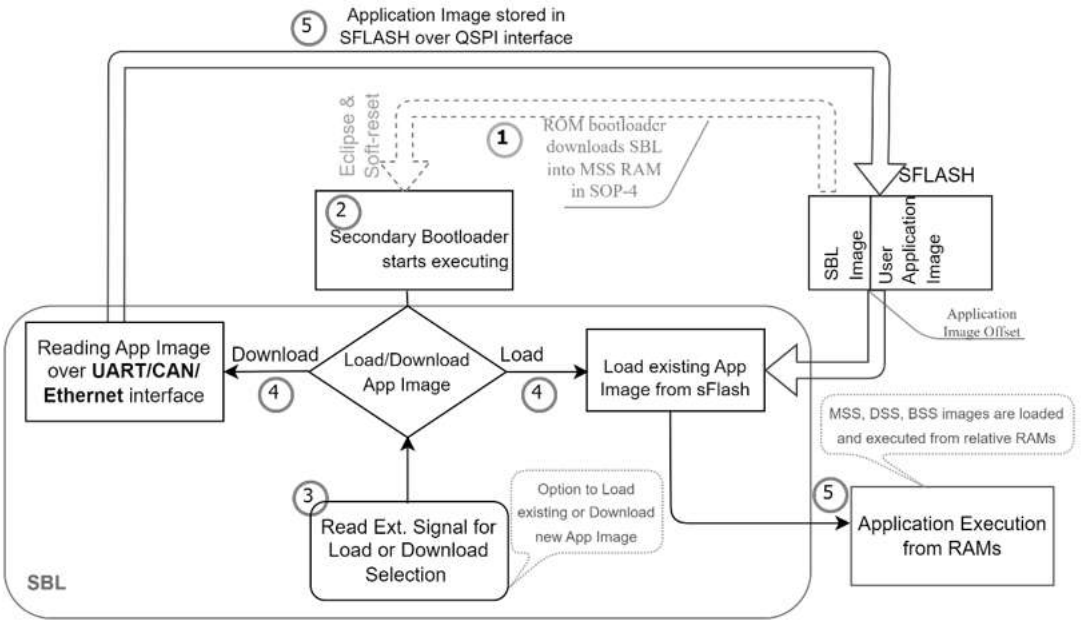

图 3-5. 从 SFLASH 加载 SBL 和应用元映像

备注

请注意，CAN 接口不适用于 AWR2544 器件。

要点

- RBL 根据 SDF 发布的用以响应 SFDP 命令的最高功能模式（四路、双路或单路）执行从 SDF 读取数据的操作。
- 对于支持四路模式的 SDF 型号，将发出四路模式命令；如果未设置四路使能 (QE) 位，则通信将失败。在此类情况下，加载流程假定 SDF 中的 QE 位已设置。
- SBL 是用户实现的实体，具有实现上述流程所需的逻辑。如果需要，SBL 可以使用 UART、CANFD 或以太网中的任何接口。
- 回退映像：如果 SDF 中的某个映像损坏，作为回退机制，RBL 仅支持从以下位置加载映像。映像的位置为：
  - META IMG1（SDF 偏移 – 0x0）
  - META IMG2（SDF 偏移 – 0x40000）

3.3.1.2 引导模式 UART

UART 引导模式基于 XMODEM 协议。可通过以下参数配置 UART 端口：

参数	值
物理端口	0
波特率	115200
数据位数	8
停止位数	1
奇偶校验	无
流控	无

R5 会启动 Xmodem 接收协议，每 3 秒（Ping 超时时间）发出一次 PING 字符（“C”）。

3.3.1.2.1 映像下载序列

通过将器件置于刷写模式便会进入此映像下载序列。

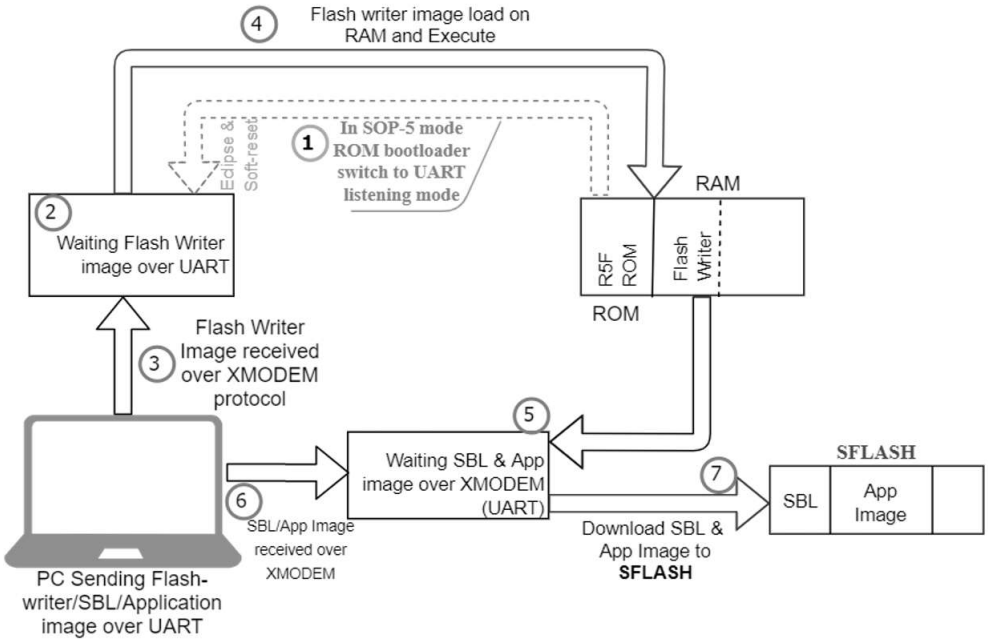

图 3-6. 加载闪存写入器映像

要点

- RBL 期望通过 UART 外设获得有效的映像。该映像可以是闪存写入器映像，然后可以通过 UART 下载 SBL 和 APP 映像并将它们存储在 sFlash 器件中。
- 如果用户希望通过 CAN 或以太网外设下载 SBL 和 APP 映像，则外设可以加载另一个相关映像（而不是使用闪存写入器映像通过 UART 下载映像）来实现这一目的。

3.4 SBL 引导

SBL 本质上是引导加载程序库的示例应用程序。由于 SBL 由主引导加载程序 RBL 进行引导，因此 SBL 被称为辅助引导加载程序。SBL 通常会执行一组特定于 SOC 的初始化，然后继续加载应用程序。

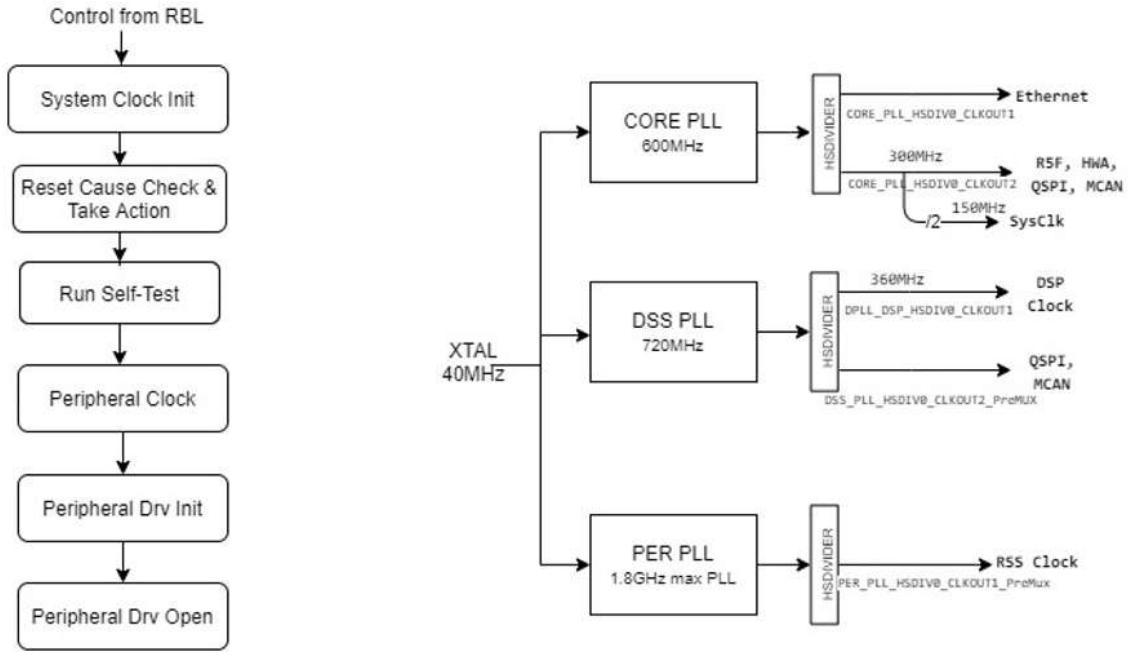

图 3-7. SBL 设计流程

辅助引导加载程序通过串行接口接收 sFLASH 中的应用程序元映像来更新该映像。然后，该引导加载程序加载并运行更新的应用程序元映像。ROM（主）引导加载程序始终加载 SBL。用户定义的 SBL 可以选择更新或加载并运行现有应用程序元映像。

SBL 可以确保出厂默认的备份映像始终不会被映像更新程序进行擦除或更新。升级过程针对整个元映像完成。用户定义的 SBL 可执行校验和验证，以验证尝试从 sFLASH 加载的元映像的有效性。在映像损坏或下载中断的情况下，SBL 提供了重新加载映像的失效防护机制。这可以通过复位电路板来实现。如果主元映像加载失败，SBL 将加载出厂默认的备份映像。如果两者都失败，SBL 可以复位电路板，以便用户可以重新尝试下载元映像。

mmWave MCU Plus SDK 提供了 SBL 的参考设计，供用户参考并编写自己的 SBL。该 SBL 的基本功能如下：

- SBL 在引导介质中的指定位置查找应用程序二进制文件的多核 appimage[GJ3]。
- 如果找到多核 appimage，则该 appimage 将解析为多个 RPRC[GJ4]。这些是经过优化的二进制文件，稍后会加载到各个 CPU 中。
- 每个 RPRC 映像都包含有关要加载映像的内核、入口点和该应用程序二进制文件的多个段的信息。
- SBL 使用此信息来初始化每个具有有效 RPRC 的内核。然后，SBL 根据指定的段加载 RPRC，设置入口点，并从复位中释放内核。现在，内核开始运行。

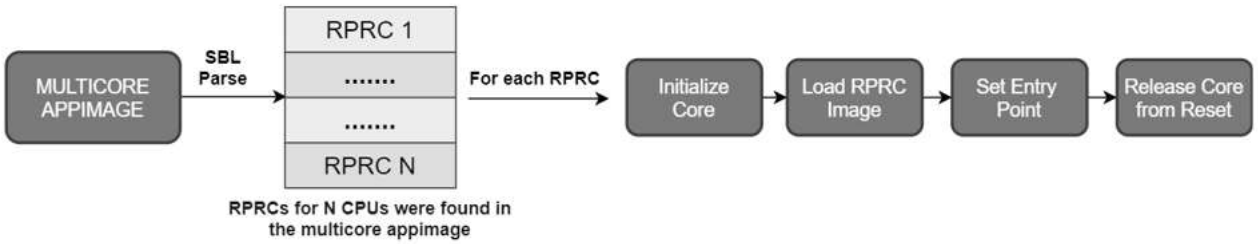

图 3-8. SBL 引导

3.4.1 R5 SBL 闪存偏移

RBL 从以下位置加载主 SBL 映像和辅助 SBL 映像：

SBL 映像	闪存偏移
主	0x00000000
辅助	0x00040000

在 QSPI 闪存模式下，首先检查主闪存偏移地址以获取有效的证书和映像。如果 R5F SBL 不存在，则会检查辅助闪存偏移以获取有效的证书和映像。如果均未找到，RBL 将在 WFI 模式下等待。HSM 引导 ROM 中的看门狗会到期 (180 秒)，系统将复位。

如果找到有效的 R5F SBL，RBL 会将其加载到 MSS L2 存储器，从 ROM 切换到 RAM，并进行引导。

3.4.2 R5 SBL 映像大小

SBL 映像 (无证书) 的最大大小为 952KB。AWR294x 器件的 RBL 只能加载 R5 L2 存储器的内容，其大小为 960KB，因此 SBL 的最大大小也是 960KB。对于具有证书的 AWR2544 器件，最大 SBL 大小为 898KB，不具有证书的器件为 890KB。

4 结论

本应用手册介绍了 RBL 引导流程和典型的 SBL 引导流程。用户可以根据文档中提到的建议来实施其 SBL。RBL 始终需要签名的映像才能工作。在 RBL 的上下文中，接收到的每个映像都称为 SBL。在功能模式下，RBL 在零偏移处从 SDF 中读取数据，并期望 SBL 已经存在于该偏移处。在刷写模式下，RBL 通过 UART 将闪存写入器映像加载到 RAM 中并开始执行。对于 RBL，也可以对该映像进行签名，并且 RBL 还会将该映像视为 SBL。因此，在这两种模式下，RBL 都会将有效映像（对于 SBL 而言）加载到 R5F 或主子系统的 RAM 存储器中。SBL 或闪存写入器或加载到 RAM 存储器中的任何其他映像的最大大小可以为 960KB（含随附的证书）。TI 还将这些不同的映像作为 SBL 进行处理。因此，示例闪存写入器映像在 MCU Plus SDK 包中名为“sbl_uart_uniflash”，只会将映像下载到串行闪存中。这些下载的映像可能是另一个 SBL 映像和应用程序映像。因此，在功能模式下，RBL 从 sFLASH 加载有效映像时，RBL 会加载先前下载的 SBL。然后，该 SBL 可以执行第 4.4 节中所述的所有必要功能，并接着将应用程序映像加载到相应的内核中。

5 修订历史记录

注：以前版本的页码可能与当前版本的页码不同

Changes from Revision * (April 2023) to Revision A (April 2024)	Page
• 通篇：将 AWR294x 更改为 AWR294x/AWR2544	0
• 将块文本从数据握手存储器更改为 DSS_MAILBOX	3

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