AN12419

适用于 i.MXRT10xx 的 Secure JTAG

Rev. 1 — November 2019 Application Note

原文链接：https://www.nxp.com/docs/en/application-note/AN12419.pdf

1 简介

本文介绍了如何使用 i.MX RT10xx MCU 系列上的 Secure JTAG。

i.MX RT 系列的系统 JTAG 控制器（SJC）提供了调节 JTAG 访问权限的能力。i.MX RT 系列提供了三种 JTAG 安全模式：

- 禁止调试模式—该模式提供最大的安全性。所有对安全性敏感的 JTAG 功能会被永久阻止，禁止任何调试。
- Secure JTAG 模式—该模式提供了高安全性。基于密钥的质询/响应认证机制用于 JTAG 访问。
- JTAG 启用模式—该模式提供了较低的安全性。这是 SJC 的默认操作模式。

此外，您还可以完全禁用 SJC 功能。为了配置这些 JTAG 模式，可以使用一次性可编程（OTP）eFuse，并在打包后进行熔断。该熔断过程是不可逆的。无法将其恢复到原始状态。在此说明，本文中使用了 Secure JTAG 模式。 目的是允许返厂/现场测试。在此模式下，允许重新激活 JTAG 端口。在硬件方面，必须将 JTAG 信号引出，并且可在应用中访问该信号。

目录

2 i.MX RT10xx Secure JTAG 支持机制

通过使用基于质询/响应的身份验证，可以在 Secure JTAG 模式下限制 JTAG 访问。任何对 JTAG 端口的访问都会经过内部验证。只有经过授权的调试设备（具有正确的响应）才能访问 JTAG 端口，否则 JTAG 访问会被拒绝。可以使用外部调试器工具（例如 SEGGER J-Link，Lauterbach Trace32，Arm RVDS / DS5 等），这些工具要求支持基于质询/响应的身份验证机制。Secure JTAG 模式通常在工厂生产中启用，而不应用于开发过程。

2.1 如何将芯片置于 Secure JTAG 模式

Secure JTAG 功能仅在 **SJC** 模式下可用，并且可以通过 JTAG_MOD 输入引脚（GPIO_AD_B0_08）进行选择。要启用 SJC 模式，必须将引脚连接到 log.1，这意味着在 CM7 DAP 模式下 Secure JTAG 不可用。

表 1. JTAG_MOD 引脚设置

Signal	Description	Pad	Mode	Direction
JTAG_MOD	SJC 模式选择。在 TRST 复位时，对该引脚进行采样，以确定 TAP 连接配置的两种可能模式。	GPIO_AD_B0_08	ALT0	IO

NOTE

由于 i.MX RT10xx EVKB 上存在的硬件冲突，可能需要对 JTAG_TDO / TDI 信号进行小的 PCB 修改，否则无法进行 JTAG 模式的通信。有关所需更改，请参考 EVKB 原理图。

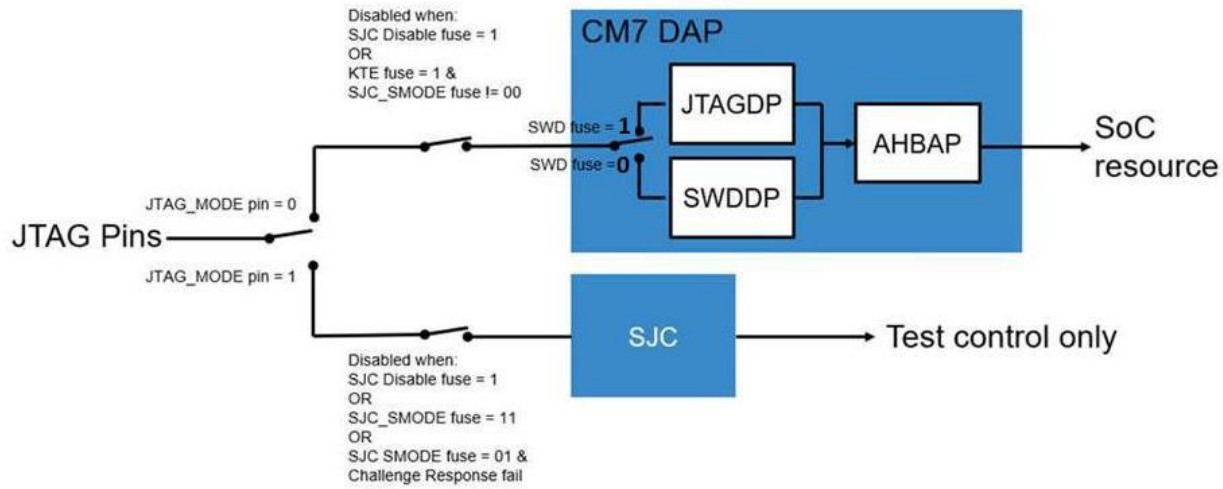

图 1. JTAG_MODE 引脚配置

2.2 i.MX RT SJC 安全模式

i.MX RT10xx 的系统 JTAG 控制器（SJC）支持三种不同的安全模式。启用 JTAG 是 SJC 的默认操作模式。用户可以通过将值 0x1 编程到标有 JTAG_SMODE 的 eFuse 中来选择 Secure JTAG 模式，如表 2 所示。eFuse 的默认值为 0x0，这意味着默认情况下 JTAG 控制器是不安全的。有关 eFuse 的更多详细信息，请参见 www.nxp.com 上，SRM_RT10xx 中的 Fusemap 和片上 OTP 控制器（OCOTP_CTRL）章节 Security Reference Manual for the i.MX RT1050 。

为了锁定并防止更改 JTAG_SMODE eFuse，除了对 JTAG_SMODE eFuse 进行编程之外，用户还应该对 BOOT_CFG_LOCK eFuse 进行编程。

NOTE

对这些熔断器进行编程，会禁用对功能和 JTAG 安全模式熔断位的访问。一旦确定了最终的熔断配置，用户应最后对其编程。建议至少将 fuse 设置为 Override Protect（OP）模式。

表 2. 与 Secure JTAG 功能相关联的 eFuses

Addr[bits]	Fuse Name	Fuse Function	Settings	Locked By
0x460[27]	JTAG_HEO	JTAG HAB 启用了 Override。 禁止启用 HAB JTAG。 通常，HAB 可以通过 OCOTP SCS 寄存器中的 HAB_JDE-bit 来启用 JTAG 调试。 JTAG_HEO-bit 可以 override 此行为。	0- HAB 可启用 JTAG 调试权限。 1- HAB JTAG 启用权限将被覆盖。 （HAB 可能无法启用 JTAG 调试权限）	BOOT_CFG_LOCK
0x460[26]	KTE	终止跟踪权限。 在 ETM 和其他模块上启用跟踪功能。	0 – 允许总线跟踪。 1 – 允许总线跟踪。 前提是符合 Secure JTAG 定义的安全状态。 （例如，JTAG_ENABLE 或 NO_DEBUG）	BOOT_CFG_LOCK
0x460[23:22]	JTAG_SMODE[1:0]	JTAG 安全模式。 控制 JTAG 调试接口的安全模式。	00 - JTAG 启用模式（默认） 01 - Secure JTAG 模式 11 – 禁止调试模式	BOOT_CFG_LOCK
0x460[20]	SJC_DISABLE	Additional JTAG 模式 具有最高级别 JTAG 保护，可以覆盖 JTAG_SMODE eFuse。 在此模式下，所有 JTAG 功能都被禁用，包括 Secure JTAG 和边界扫描。	0 – 启用 JTAG 1 – 禁用 JTAG	BOOT_CFG_LOCK
0x460[19]	DAP_SJC_SWD_SEL	控制 DAP 在 JTAG 模式或 SWD 模式下工作。	0 - DAP 在 SWD 模式下工作 1 - DAP 在 JTAG 模式下工作	BOOT_CFG_LOCK
0x400[3:2]	BOOT_CFG_LOCK[1:0]	对与 BOOT 相关的 fuse 进行锁定保护。 该 fuse 可锁定众多功能，包括 JTAG_SMODE。	00 - Unlock 1x - Override Protect (OP) x1 - Write Protect (WP) 11 – 同时具备 OP 和 WP	N/A
0x400[6]	SJC_RESP_LOCK	SJC 响应锁定	0 - Unlock 1 - Lock (WP,OP,RP, sense)	
0x600	SJC_RESP[55:0]	Secure JTAG 控制器的响应参考值		SJC_RESP_LOCK （读和探测都会被锁定）

NOTE

安全级别不能降低，只能提高。由于调试模式由 OTP（硬件熔断器）控制，因此熔断位只能烧毁一次。

例如，可以对有关模式进行以下更改：

– 从“JTAG Enabled” 到 “Secure JTAG”

– 从“Secure JTAG” 到 “No debug”

2.3 Secure JTAG eFuses

用于认证 JTAG 访问权限的质询/响应机制，使用质询值和关联的响应密钥。密钥存储在 IC 内部的 eFuse 中。下面列出了 i.MX RT1050 系列中，用于存储质询值和响应密钥的 eFuse：

- 质询值是已编程到 eFuse 中的“设备唯一 ID”。每个 IC 的设备 ID 都是唯一的，可以从 OCOTP 寄存器 HW_OCOTP_CFG0 和 HW_ OCOTP_CFG1 中读取。eFuse 在制造过程中就完成了编程。
- 用户将响应密钥（56 位）编程到标记为 SJC_RESP 的 eFuse 中。
- 需要对 KTE fuse 进行编程，以使其具备 JTAG 安全模式。每个 POR 只能输入一次响应码。如果响应码不正确，则用户必须先复位芯片的 POR，然后才能尝试其他响应码。 POR 会清除 SNVS 区域以外的敏感数据。

对响应密钥进行编程之后，对于 Arm 核上运行的软件，用户必须禁用其读取或覆盖响应密钥的功能。通过将 0x1 编程到关联的 lock eFuse HW_OCOTP_LOCK_SJC_RESP，可以完成此步骤。

响应值由用户决定。一旦配置并锁定了响应 fuse 字段，Arm 核将无法读取该值。

2.4 SW 启用 JTAG

通过向 efuse 控制器模块中的 HAB_JDE（HAB JTAG DEBUG ENABLE）写入 “1”，可以在 SW 中绕过 Secure JTAG 身份验证。通过此方法，可以打开 JTAG，无论其安全模式如何。S/W JTAG 启用允许 JTAG 启用，而无需激活质询-响应机制。

在将控制权转移到应用程序代码之前，平台初始化软件应该为 JDE bit 设置 LOCK bit，以确保只有受信任的 SW 才能设置 JDE 位。

JTAG SW 启用不允许在启动或内存故障时进行调试，因为在进入调试之前它需要进行复位。

可以通过烧毁 JTAG_HEO 熔断器来永久禁用 JTAG_JDE bit SW 后门访问权限。

NOTE

S/W 启用的 JTAG 功能依赖于 S/W 保护，从而降低了系统的整体安全级别。如果不需要此功能，强烈建议熔断 JTAG_HEO e-fuse 以禁用此功能。

2.4.1 HAB（高安全启动）中的 JDE 位控制

通过 Authenticate CSF 命令解锁后，可以通过 ROM 启动 SW 将 HAB_JDE 设置为 “1”。

在生成具有签名的程序镜像之前，用户必须编辑.sb 文件中的 UNLOCK 部分，并以 8 字节序列的正确格式为设备提供特定的 UID，请参见下面的 UID = 0x63e1841b440b81d2 示例，请看：

```
section (SEC_UNLOCK;
Unlock_Engine = "OCOTP",
Unlock_features = "JTAG, SCS, SRK REVOKE",
Unlock_UID = "0xe1, 0x63, 0x1b, 0x84, 0x0b, 0x44, 0xd2, 0x81"
```

```
)
```

有关 HAB（高安全启动）中，由平台初始化 SW 控制 HAB_JDE SW 的更多信息，请参阅[5]中的 **5.2.13 Unlock (HAB only)**。

2.5 Secure JTAG 的 debug 认证协议

当 SJC 处于 Secure Debug 模式时，身份认证过程如下：

1. JTAG 通过测试数据输出（TDO）链调用质询密钥。
2. 在主机端，debug 工具将质询密钥作为输入并生成相应的响应密钥。
3. 通过测试数据输入（TDI）链，对应的响应密钥被调回。
4. SJC 将内部的响应密钥与调回的密钥进行比较，并仅在密钥匹配时启用 JTAG 访问。

NOTE

JTAG 访问授权之后的任何设备重置都会将 JTAG 控制器切换回其锁定状态。

图 2 展示了质询/响应机制如何与 JTAG 工具一起运作。

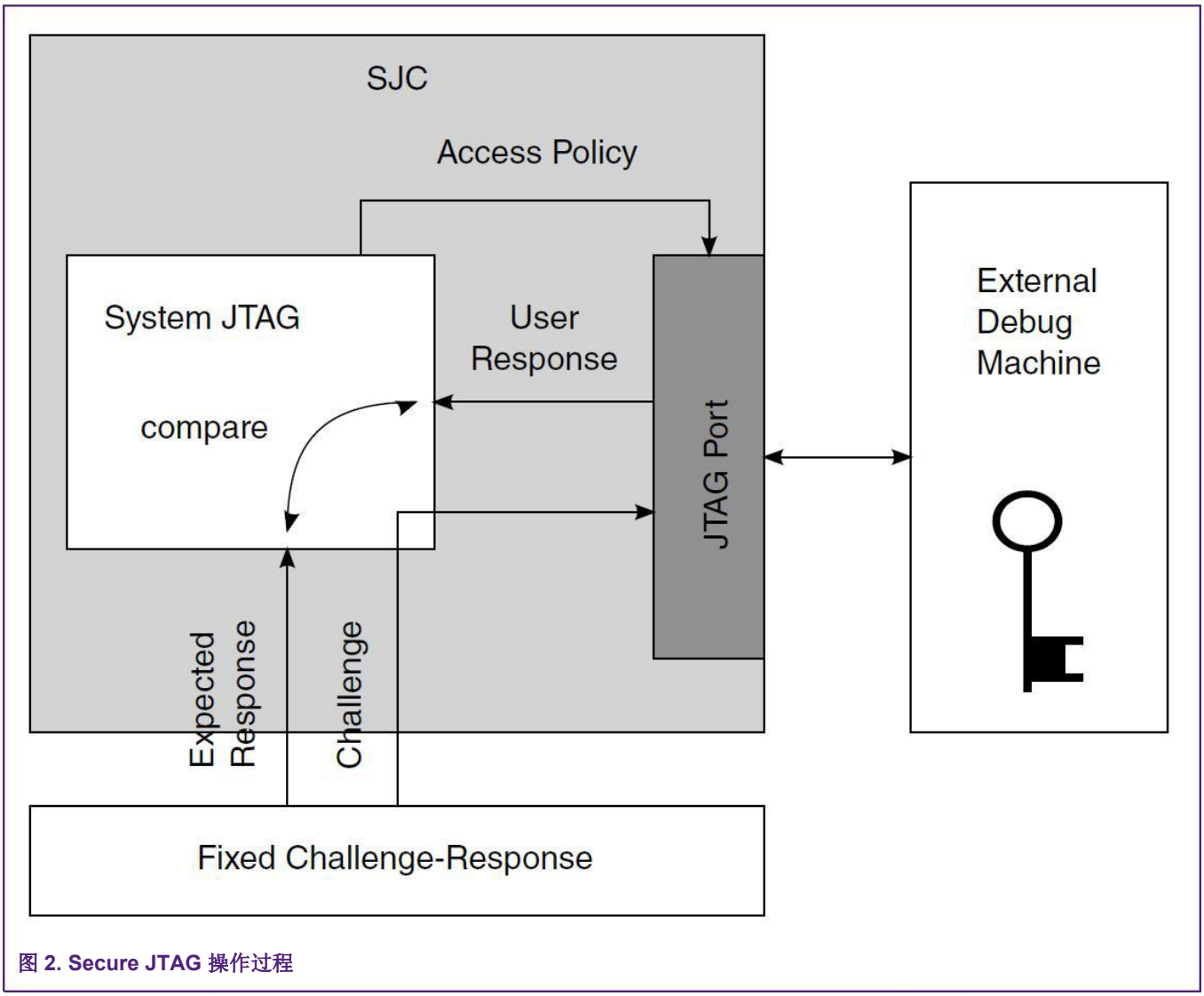

图 2. Secure JTAG 操作过程

JTAG 调试工具将检索到的质询密钥传递给用户的应用程序，并返回对应的响应密钥。质询/响应的密钥对管理由用户决定，不由 NXP 或调试工具供应商决定。Secret response key approaches 中进一步讨论了密钥管理。

2.6 SJC 禁用 fuse

在 SJC 内部，除了可以实现各种 JTAG 安全模式之外，还可以通过 SJC_DISABLE eFuse 禁用 SJC 功能。该 eFuse 创建了一个附加的 JTAG 模式，即 JTAG Disabled（禁用 JTAG），它具备最高级别的 JTAG 保护，覆盖了 JTAG_SMODE eFuse。在这种模式下，所有的 JTAG 功能都被禁用，包括 Secure JTAG 和边界扫描。用户如果需要使用 Secure JTAG 功能，必须确保熔断器不会烧毁。

3 响应密钥方法介绍

对于使用 JTAG 指令检索到的质询值（i.MX RT10xx 中的“设备唯一 ID”），都有一个仅用户知道的关联响应密钥。JTAG 工具供应商仅能控制身份验证过程使用的 JTAG 机制，并不知道已编程到 eFuse 中的响应密钥值。是由用户来确定所采用的保护级别。

以下是用户的应用程序用于响应密钥管理的策略。

1. 相同的响应密钥——每个芯片使用相同的响应密钥。用户可以选择所有的芯片烧写同样的相应密钥。从安全角度来看，这是最简单且最不复杂的用法。如果未经授权的用户可以访问烧写过的响应密钥，则可以通过 JTAG 端口访问与该响应密钥相同的所有产品。
2. 唯一响应密钥的数据库——用户保有一个数据库，其中包含生成的所有响应密钥。用户应用程序可以基于质询值进行查表。可以建立一个安全的服务器，其中包含对用户进行身份验证的质询/响应密钥对，但这需要独立运作。必须读取所有 IC 的质询值，并且必须建立一个用于匹配质询响应对的数据库。存储和管理大量响应密钥并非易事，但从安全角度来看是有利的，因为它不依赖于任何易破解的算法。
3. 通过算法生成的响应密钥——响应密钥是基于算法生成的。使用此方法，无需管理大型数据库。例如，该算法可以使用质询值来生成响应密钥。该响应密钥已编程到 SJC_RESP eFuse 中。然后，每次通过 JTAG 检索质询值时，用户应用程序都可以对其进行处理，并可以为 JTAG 调试工具生成对应的响应密钥。一旦公开了算法或对算法进行了逆向工程，该方法就不再安全。

NOTE

NXP 不提供安全响应密钥管理或密钥生成服务；这些内容不在本文档的范围之内。

3.1 使用 NXP 工具对 Secure JTAG eFuse 进行编程

要在芯片上编程 Secure JTAG 所需的 eFuse，用户应首先遵循以下步骤。有关片上 OTP 控制器（OCOTP_CTRL）和 Fusemap 的信息，请参见 www.nxp.com 上相应的 i.MX RT10xx 系列参考手册。以下步骤使用 NXP MCU Boot Utility 对 eFuse 进行编程。

1. 从该网站下载最新的 NXP MCU Boot Utility：http://www.nxp.com
2. 用户应将以下数值编程到 Secure JTAG 所需的 eFuse：
   - 读取并备份：存储在 eFuse UUID [1,0]，位置（0x420、0x410）上的 64 位“质询”值。参见图 3。
   - 在 eFuse SJC_RESP 的位置（0x610、0x600）中编程一个 56 位（7 字节）的响应密钥。在下面的示例中，值“0xedcba987654321”将被编程。

NOTE

在 图 3 中， MSB 存储在较高位的地址，而最左边的字节为 0x00，会被忽略。

用户应定义自己的响应密钥，并保留密钥备份以备将来使用。

   - 在 eFuse DAP_SJC_SWD_SEL 中编程 0x1 以将 DAP 切换到 JTAG 模式。
   - 在 eFuse JTAG_SMODE 中编程 0x1，以将 SJC 切换到 Secure JTAG 模式。
   - 在 eFuse KTE_FUSE 中编程 0x1。
   - 最后，用户必须在 eFuse SJC_RESP _LOCK 中编程 0x1，以禁用对响应密钥的读/写访问。在该操作之后，秘密响应字段“SJC_RESP”在 fuse map 中变为“不可见”。 参见 图 3 和 图 4 。

下图演示了如何使用 NXP MCU Boot Utility 对上述 eFuse 进行编程。eFuse operation utility 是该工具的一部分，可以读取和写入 eFuse 映射寄存器。进行写入（刻录）操作时请小心，因为这是不可逆的，并且可能会完全锁定设备或某些功能。

要启用 Secure JTAG，请按照上述 使用 NXP 工具对 Secure JTAG eFuse 进行编程 提及的步骤进行操作，有关 eFuse bits 的更多详细信息，请参见 表 2，图 3 和 图 4 。

该示例未对 BOOT_CFG_LOCK [1：0]和 eFuse 进行编程，以防止对 JTAG_SMODE eFuse 进行更改。在对这些锁定的 eFuse 进行编程时，除了禁用 JTAG 模式位，还应禁用对功能的访问。因此，一旦确定了最终配置，便应执行该操作。

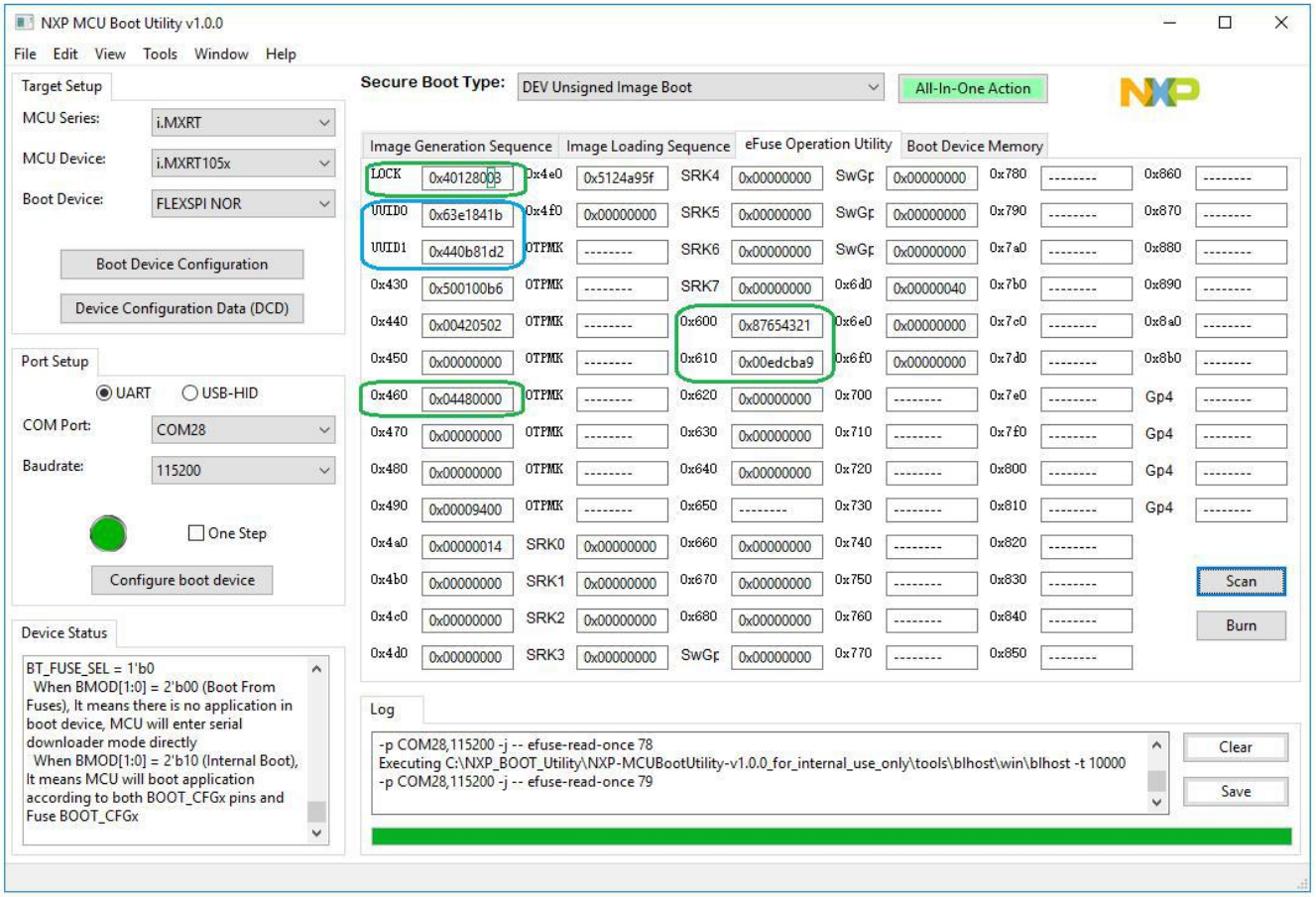

图 3. eFuse Operation Utility – 安全响应密钥配置

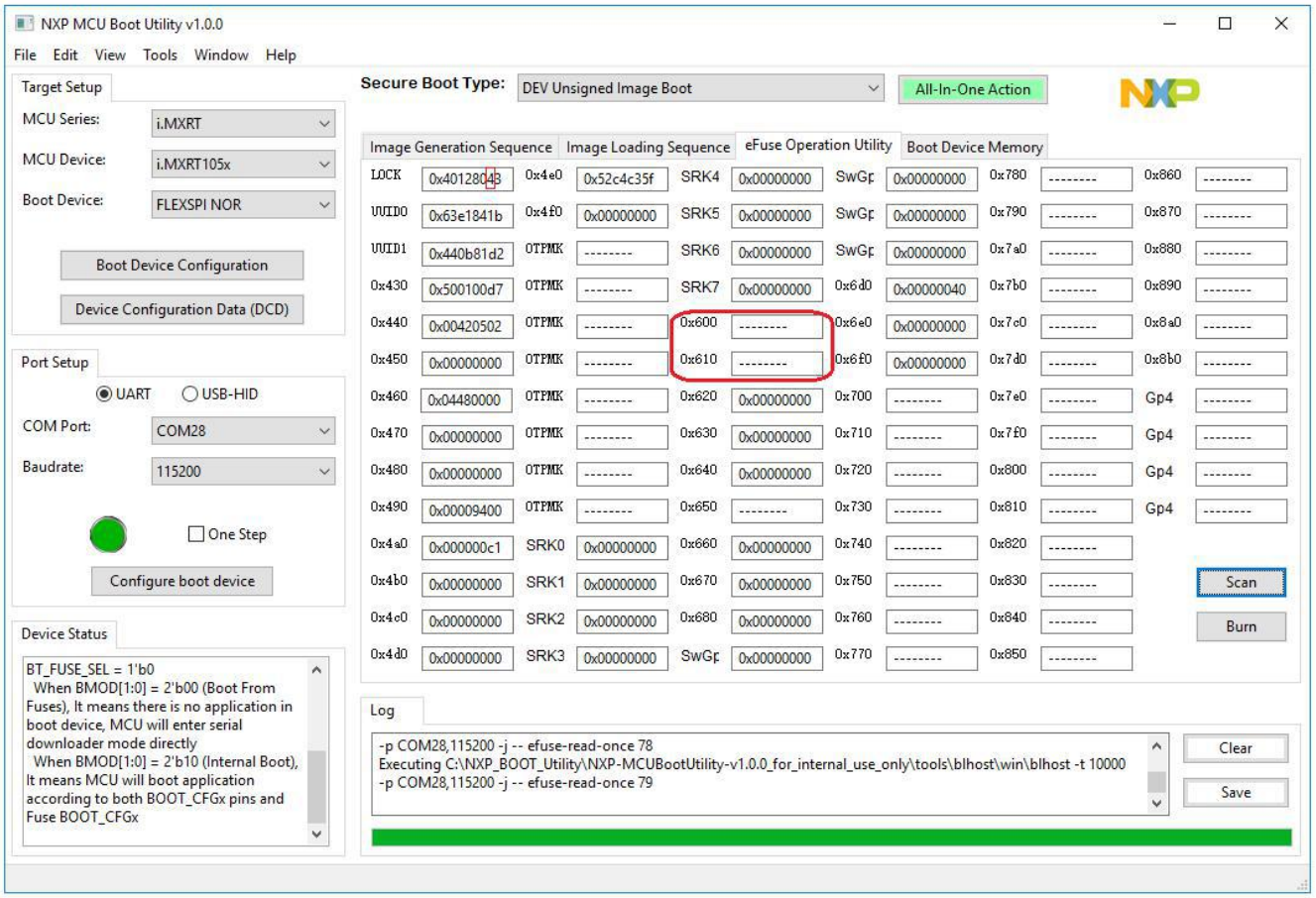

图 4. eFuse Operation Utility – 安全响应密钥锁定

4 启用 Secure JTAG 进行调试

要使用 Secure JTAG 功能，JTAG 调试器必须支持该功能。本节提供的示例使用 SEGGER J-Link 调试工具。

尽管以下示例中使用了 IMXRT1050-EVKB 板上的 i.MX RT1052 设备。它们也可以应用于其他 RT10xx 器件。以下步骤假定用户具有使用调试工具和 NXP MCU Boot Utility 的经验。

4.1 通过 Secure JTAG 连接 J-Link 调试器的步骤

使用 Secure JTAG 时，以下步骤将 SEGGER J-Link 调试工具连接到 i.MX RT10xx：

1. 下载 SEGGER J-Link 软件和文档包：

   https://www.segger.com/downloads/jlink/#J-LinkSoftwareAndDocumentationPack

   如果您需要从 SEGGER 主页寻找相关脚本进行参考，它们位于*“Downloads” - “J-Link / J-Trace” - “J-Link Software and Documentation Pack”*.

2. 下载并编辑名为 NXP_RT1052_SecureJTAG.JlinkScript 的 J-Link 脚本文件。可以请求从 NXP 得到脚本文件。在此文件中，将已编程的响应密钥添加到 SJC_RESP eFuse 中。在以下示例中，响应密钥为“0xedcba987654321”，并与 使用 NXP 工具对 Secure JTAG eFuse 进行编程 中提及的 eFuse 中编程的响应密钥进行匹配。

// Secure response stored @ 0x600, 0x610 in eFUSE region (OTP memory)

Key0 = 0x87654321;

Key1 = 0xedcba9;

3. 使用 log.1 中 JTAG_MODE 引脚（GPIO_AD_B0_08）对板子进行供电或复位。用户必须手动执行此操作，因为信号没有连接到 EVB 上的调试连接器。
4. 找到 SEGGER SW J-Link 的安装目录。
5. 使用之前提及的脚本文件作为参数运行 “jlink.exe”。

   例如：

   jlink.exe -JLinkScriptFile NXP_RT1052_SecureJTAG.JlinkScript -device MCIMXRT1052 -if JTAG -speed 4000 - autoconnect 1 -JTAGConf -1, -1

NOTE

外部 IDE 工具可以使用相同的脚本文件调用 “JLinkGDBServer.exe”应用程序，以取消对目标的保护。

工具脚本应从 eFUSE UUID [1,0]位置读取质询值。并且，它为 SJC 的身份验证匹配提供了适当的响应。

JTAG_MODE 引脚必须切换到 log.0，有关详细信息，请参见 图 1 和 图 6。用户应通过更改板上的引脚，手动进行此操作。如果将引脚连接到 JTAG 连接器，并且该工具支持对信号的控制，则该工具可以自动执行此操作。

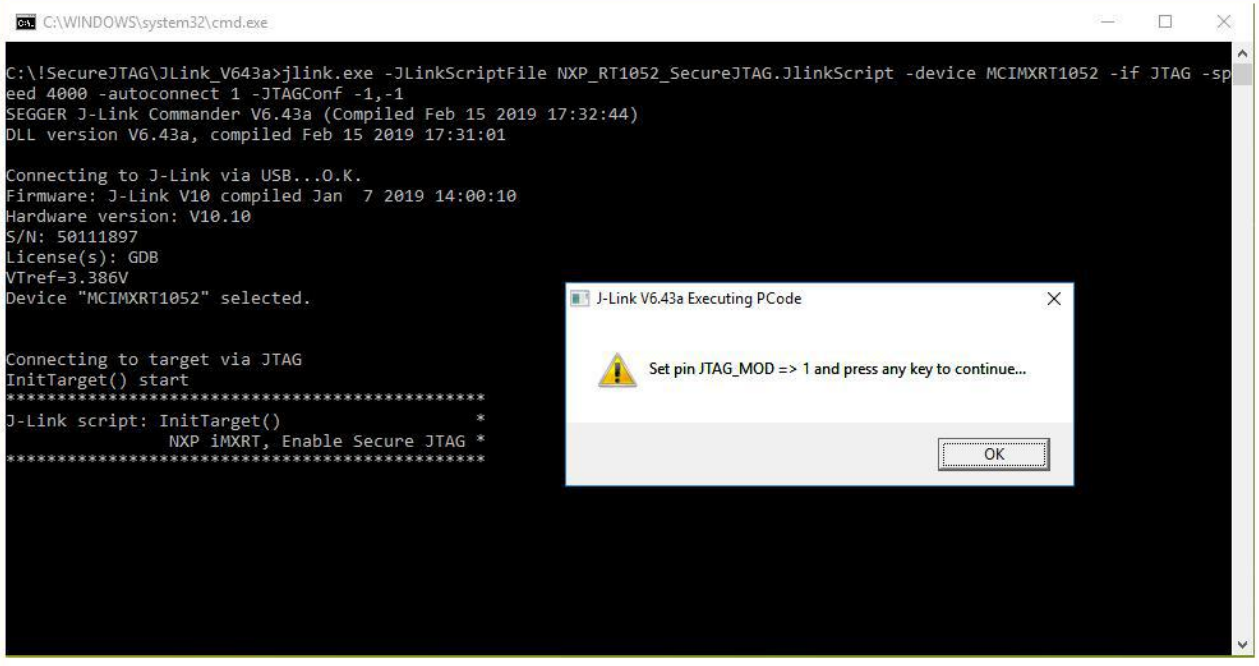

图 5. 在受保护的 i.MX RT 设备上运行 SEGGER J-Link 脚本

调试工具通过 JTAG 可以成功连接到 i.MX RT10xx 目标。图 7 中的截图展示了通过 Secure JTAG 成功进行的连接：

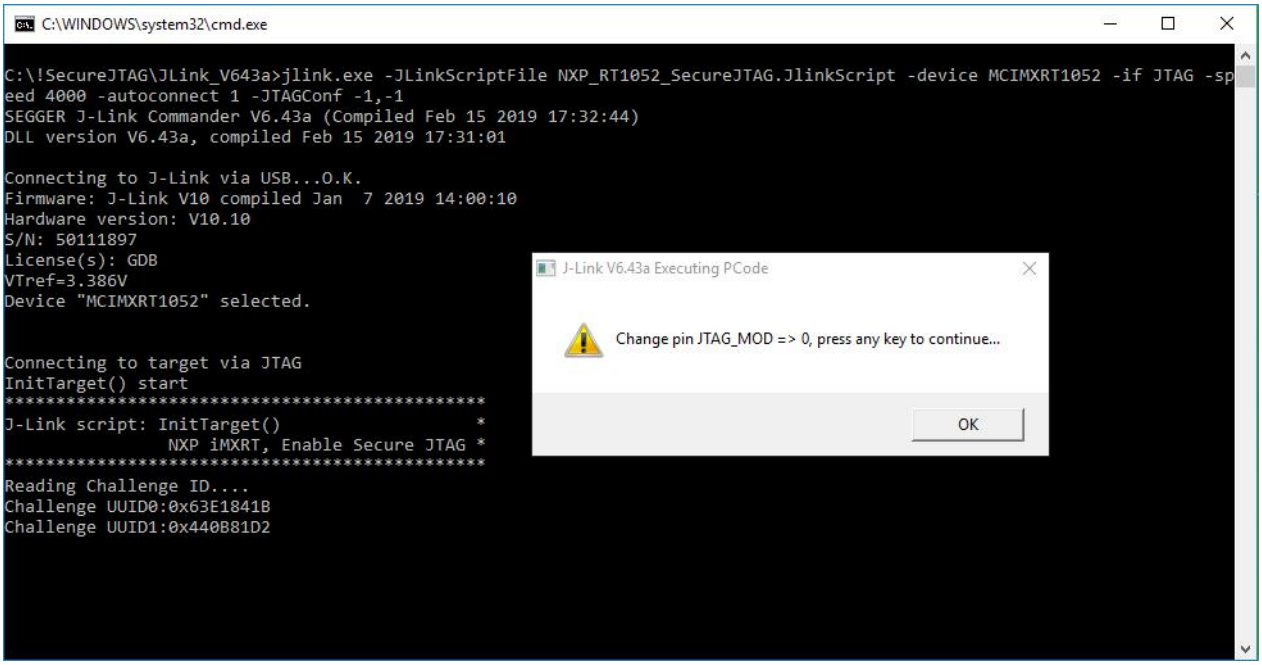

图6. 用SEGGER J-Link脚本读取质询ID

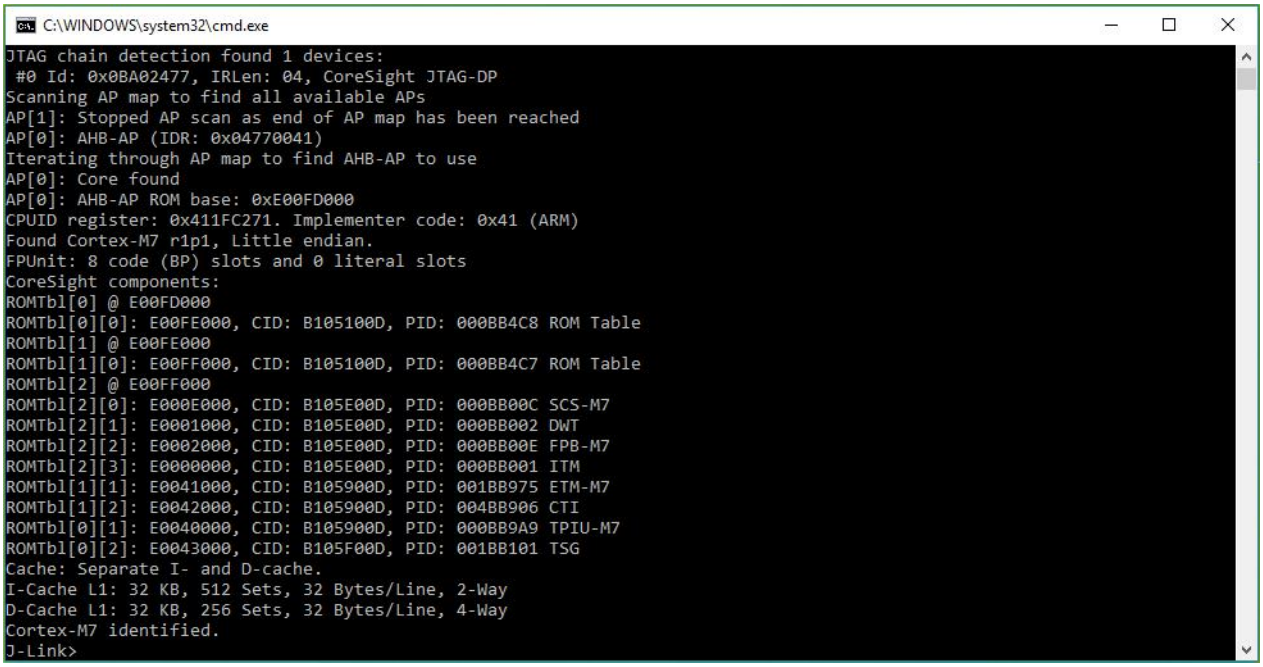

图 7. SEGGER J-Link成功连接到安全JTA

用户现在可以执行正常的JTAG调试操作，因为已使用质询-响应机制对设备进行了身份验证。

NOTE

JTAG 访问授权后的任何复位都会将 JTAG 控制器转换到锁定状态，需要重复身份验证过程。

6. 为确保 i.MX RT 系列的 SJC 在安全模式下运行，请编辑“NXP_RT1052_SecureJTAG.JlinkScript”文件，提供错误的响应密钥，然后重新运行脚本。调试工具无法通过 JTAG 连接到 i.MX RT10xx 目标。

4.2 SEGGER J-link Secure JTAG 的解锁脚本示例

```
int InitTarget(void) {
  int v;
  int Key0;
  int Key1;

  JLINK_SYS_MessageBox("Set pin JTAG_MOD => 1 and press any key to continue...");

  // Secure response stored @ 0x600, 0x610 in eFUSE region (OTP memory)
  Key0 = 0x87654321;
  Key1 = 0xedcba9;

  JLINK_CORESIGHT_Configure("IRPre=0;DRPre=0;IRPost=0;DRPost=0;IRLenDevice=5");
  CPU = CORTEX_M7;
  JLINK_SYS_Sleep(100);
  JLINK_JTAG_WriteIR(0xC); // Output Challenge instruction

  // Readback Challenge, Shift 64 dummy bits on
  TDI JLINK_JTAG_StartDR();
  JLINK_SYS_Report("Reading Challenge ID....");
  // 32-bit dummy write on TDI / read 32 bits on
  TDO JLINK_JTAG_WriteDRCont(0xffffffff, 32);
  v = JLINK_JTAG_GetU32(0);
  JLINK_SYS_Report1("Challenge UUID0:", v);

  JLINK_JTAG_WriteDREnd(0xffffffff, 32);
  v = JLINK_JTAG_GetU32(0);
  JLINK_SYS_Report1("Challenge UUID1:", v);

  JLINK_JTAG_WriteIR(0xD); // Output Response instruction

  JLINK_JTAG_StartDR();
  JLINK_JTAG_WriteDRCont(Key0, 32);
  JLINK_JTAG_WriteDREnd(Key1, 24);

  JLINK_SYS_MessageBox("Change pin JTAG_MOD => 0, press any key to continue...");

  return 0;
}
```

5 总结

本应用指南描述了 Secure JTAG 的 eFuse 配置和身份验证过程，使用 SEGGER J-Link 脚本进行了验证和演示。其他调试工具（如 Lauterbach Trace32 和 Arm DS5）的支持机制和示例将涵盖在更新的版本中。

6 参考文献

1. Configuring Secure JTAG for the i.MX 6 Series Family of Application Processors (AN4686)

2. Security reference Manual for the i.MX RT1050 Processor (IMXRT1050SRM), available upon a request from: www.nxp.com
3. J-Link / J-Trace User Guide https://www.segger.com/downloads/jlink/UM08001>
4. Training JTAG Interface, Lauterbach TRACE32 http://www2.lauterbach.com/pdf/training_jtag.pdf
5. HAB Code-Signing Tool User's Guide (Rev. 3.2.0, 04/2019) https://www.nxp.com/webapp/Download? colCode=IMX_CST3.2.0_TOOL&location=null

7 修订历史

表 3. 修订历史

版本号	日期	修改内容
Rev. 0	04/2019	Initial release with J-Link script example
Rev. 1	11/2019	HAB_JDE bit information added

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer’s applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

For more information, please visit: http://www.nxp.com

For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: November 2019

Document identifier: AN12419

arm