AN13037
LPC55Sxx 身份鉴权调试

第 1 版 — 2021年10月6日 应用笔记

目录

1 简介

LPC55Sxx 系列器件包含了配置调试端口和调试固件的能力。由于调试功能需要访问系统状态和系统信息，而安全的定义本质则是限制对系统资源的访问，所以调试功能与安全功能原则上是相冲突的。因此，许多产品在发布之前需要完全禁用调试权限以保证安全，但是这给产品设计人员的产品质量问题导致的返厂分析（Return Material Analyze，RMA）带来了巨大挑战。为了应对这些挑战，LPC55Sxx 提供了调试身份鉴权协议（DAP）作为对调试器（外部实体）进行身份验证的机制，在授予对设备的调试访问权限之前，该调试器需要具有产品制造商授权的证书。图 1展示了调试身份验证的示例用法，OEM是根密钥对的所有者，在制造过程中，根密钥哈希值被编程到设备中。当终端客户遇到问题时，设备会被运送到维修中心，现场技术人员可以向OEM发送请求获取调试证书（DC），该证书由私有根密钥签名，现场技术人员使用此证书供调试访问。

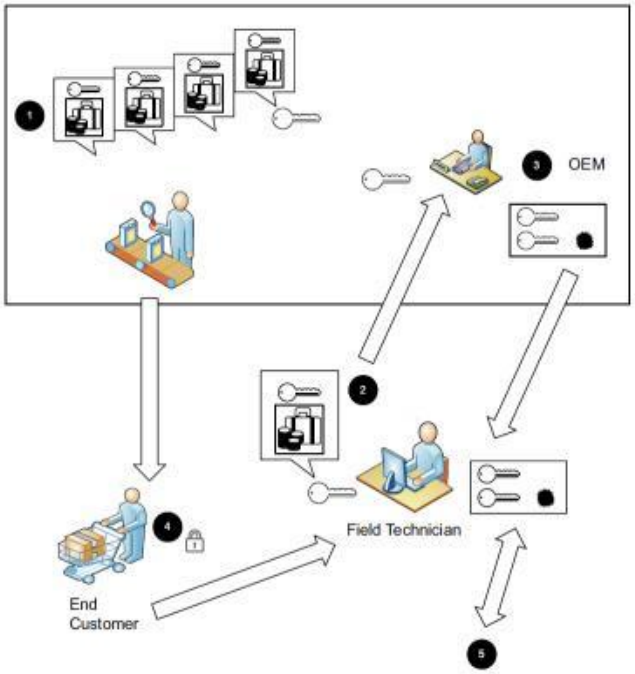

图1. 调试鉴权协议使用示例

2 概述

调试端口一般使用单线调试（SWD）接口，ARMv8-M架构明确了保护调试端口的可能性，并提供了在已锁定设备上重新打开调试端口的方法。NXP提供基于不同长度加密密钥的调试鉴权协议。

1.0 版本使用 RSASSA-PKCS1-v1_5 签名验证，使用具有 2048 位模数和32位指数的 RSA 密钥。

1.1 版使用 RSASSA-PKCS1-v1_5 签名验证，使用具有 4096 位模数和32位指数的 RSA 密钥。

LPC55Sxx上的调试身份鉴权模块确保调试器拥有所需的调试权限，当身份验证成功的话，可以通过调试接口正常访问访问受限部分。

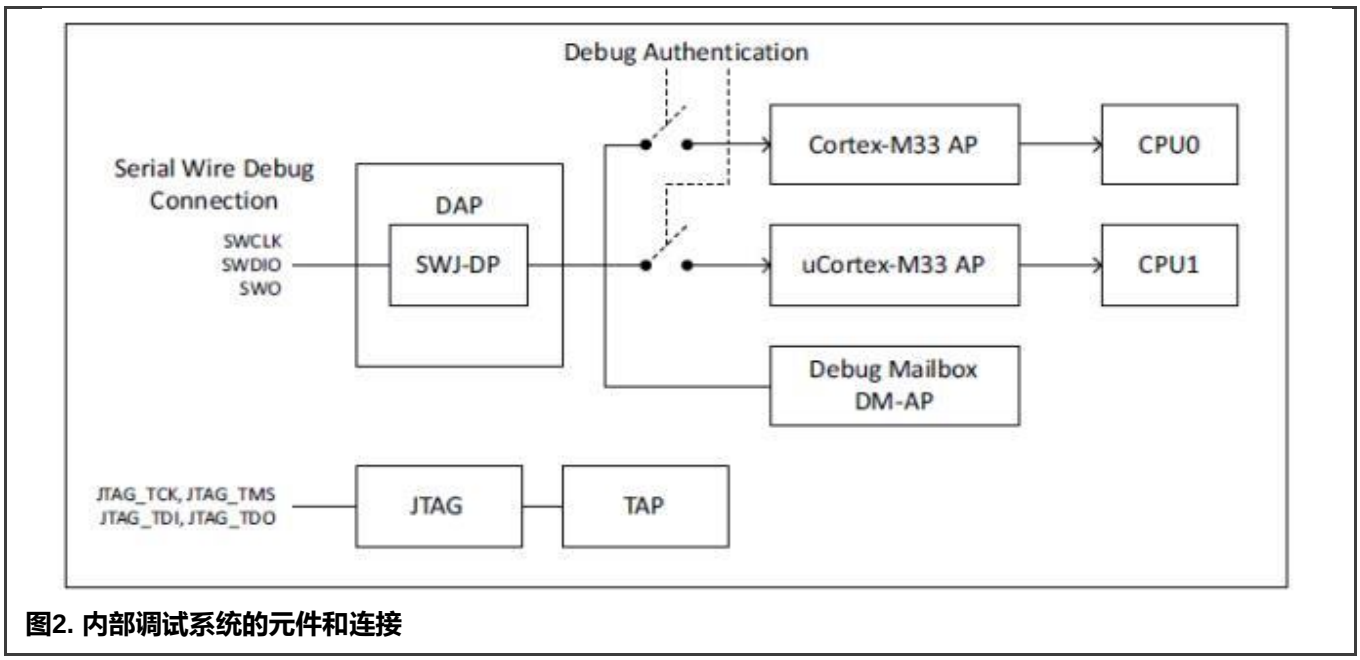

图2. 内部调试系统的元件和连接

- DAP：调试访问端口有一个串行线路端口（SWJ-DP），用于解释传入的数据并将其路由到适当的访问端口（AP）。
- CPU0 AP：实例化为 CPU0 的 Cortex®-M33 内核的调试访问端口。
- CPU1 AP：实例化为 CPU1 的 Cortex-M33 内核的调试访问端口，此 CM33 实例没有安全扩展（用于 Armv8-M 的 TrustZone）。
- DM-AP：调试邮箱（DM）的调试访问端口，DM 用于通过发送/接收消息与从 ROM 执行的代码进行通信。
  - 此端口始终启用，外部可以和 ROM 互相发送和接收数据。
  - 该端口用于实现 NXP 调试鉴权协议。

SWJ-DP 和 DM-AP 模块始终可通过 SWD 接口访问，其余模块（CPU0 和 CPU1 AP）在硬件状态机和软件控制下启用/禁用。

CPU0 的 DAP 在上电复位或复位管脚有效期间被禁用，如果当遵循正确的调试启动程序时，它会由ROM来启用。如果未使用DAP，则可以使用调试启用协议来启动调试会话。调试鉴权过程允许受控于由 Cortex-M33 生成的DBGEN、NIDEN、SPIDEN和SPNIDEN信号，具体如下所述。

DBGEN：针对 Arm8-M 定义的非安全域，对 TrustZone 进行侵入式调试。

- 用于在特定活动上停止处理器的断点和观察点。
- 用于检查和修改寄存器和内存，并提供单步执行的调试连接。

NIDEN：针对 Arm8-M 定义的非安全域，对 TrustZone 进行非侵入式调试。

- 关于指令执行和数据传输的信息集合。
- 实时跟踪并传送数据到片外工具，以将数据与开发软件上的源代码合并用作将来的分析。

SPIDEN：针对 Arm8-M 定义的安全域，对 TrustZone 进行侵入式调试。

SPNIDEN：针对 Arm8-M 定义的安全域，对 TrustZone 进行非侵入式调试。

CPU1 默认处于复位模式，复位状态必须由 CPU0 释放才允许访问CPU1 AP。CPU1（Cortex-M33 核）的调试访问端口在复位时禁用，并由硬件状态机启用。

该端口利用额外的控制来启用/禁用不同的功能，如下所述，通过 DEBUG_FEATURES 寄存器（SYSCON 中的偏移量 0xFA4）和 DEBUG_FEATURES_DP 寄存器（SYSCON 中的偏移量 0xFA8）进行控制。

调试器邮箱（DM）AP 是一个基于寄存器的邮箱，可由 CPU0 和 MCU 的设备调试端口（DP）访问。该端口始终处于启用状态，通过 SWD 接口进行通信的外部主机可以与 CPU0 的 ROM 执行的启动代码交换消息和数据，该端口用于实现NXP调试鉴权协议。

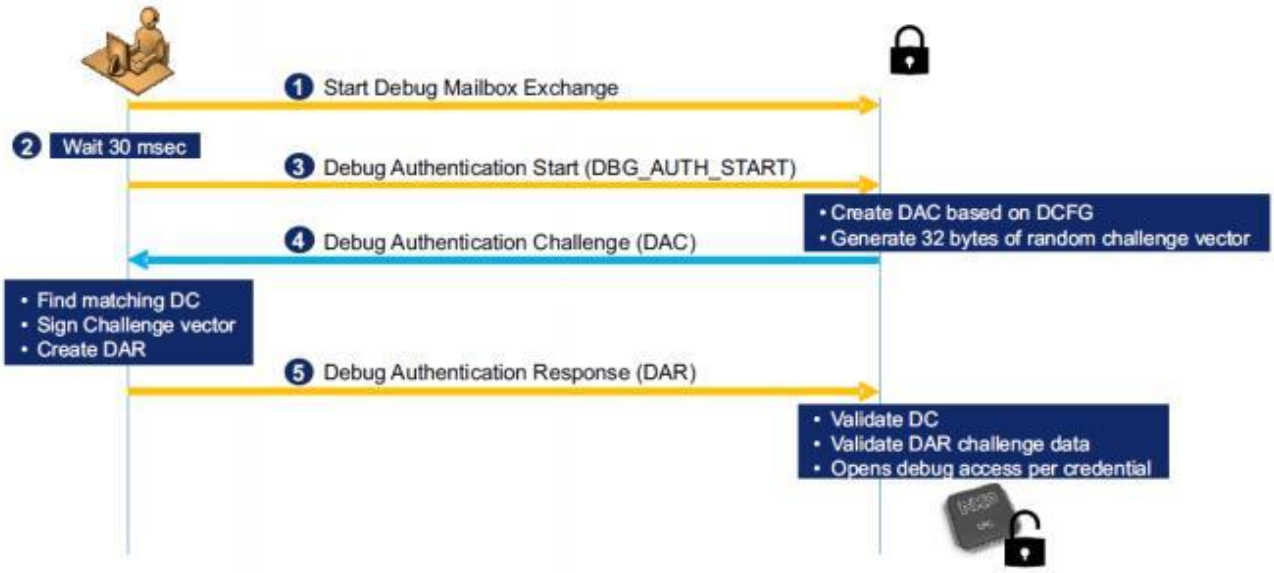

图3. 调试认证流程

3 示例程序

3.1 环境

3.1.1 硬件环境

- 板子：

  — LPC55S69EVK- Rev.A2, 1B版本。

- 调试器：

  — 板上集成 CMSIS-DAP 调试器。

- 其他：
  — 1根 Micro USB 电缆。
  — PC。
- 板级设置：
  — 在 PC 和板上 P6 之间连接 Micro USB 电缆以加载和运行演示。

3.1.2 软件环境

- 工具链：
  — MCUXpresso IDE 11.1.1或更高版本
  — Python 3.6或更新版本
  — SPSDK 1.4.0 - https://github.com/NXPmicro/spsdk
- 软件包：
  — SDK_2.8.2_LPCXpresso55S69

3.2 步骤

下面的过程描述了如何安装 SPSDK 并使用 LPC55S69 设备的示例配置，以及启用调试身份鉴权这项功能。在此示例中，除在系统编程（ISP）模式外，所有调试端口都启用了调试身份鉴权。ISP 模式一直处于启用状态。ISP 模式用于在加载错误配置的情况下进行测试。理论上是有可能重新加载配置的，当然在客户的最终配置中不会出现这种情况。该示例可以在本应用笔记的相关软件包中找到。

在设备配置时要小心。当 SHA-256 摘要被写入 CMPA 或加载了错误的配置时，将无法返回并解锁设备。

3.2.1 安装 SPSDK

1. 在您的电脑中安装 Python 3.6 或更新版本。
2. 建议在您的工作空间中创建一个 Python 虚拟环境：

   ```
   python -m venv nxp\venv
   ```

   （该示例使用 C:\nxppath 作为根文件夹）。

   启用您的 Python 虚拟环境：

   ```
   venv\Scripts\activate
   ```

   虚拟环境有效激活后，您将在命令行中看到`(venv) C:\nxp>`。
3. 将 SPSDK 安装到您的 VENV 中：

   ```
   pip install -U spsdk
   ```
4. 检查 SPSDK 是否正确安装在您的 VENV 中：

   ```
   spsdk --help
   ```

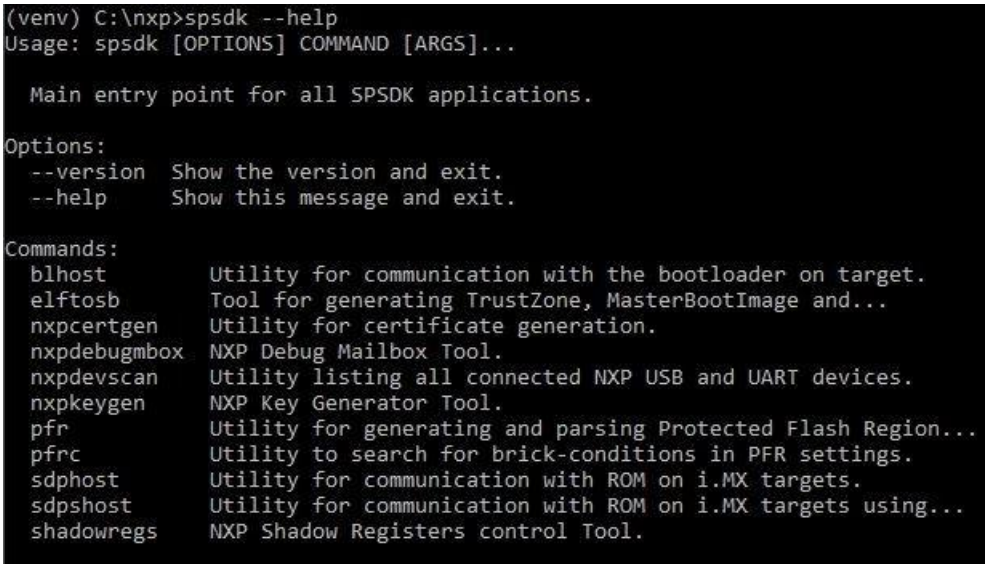

图4. SPSDK – 用于输出

3.2.2 加载/生成密钥

RSA 密钥对用于身份鉴权。私钥用于签署调试证书，可信任根公共密钥（RoTK）的哈希值存储在 MCU (PFR) 的非易失性存储器中。

对于定位已创建和使用的加密密钥、生成用于保护设备的密钥对以及为设备配置和安全性生成新的密钥对是有必要的：

```
openssl genrsa -out rotk0_rsa_2048.pem 2048
```

```
openssl rsa -in rotk0_rsa_2048.pem -pubout > rotk0_rsa_2048.pub
```

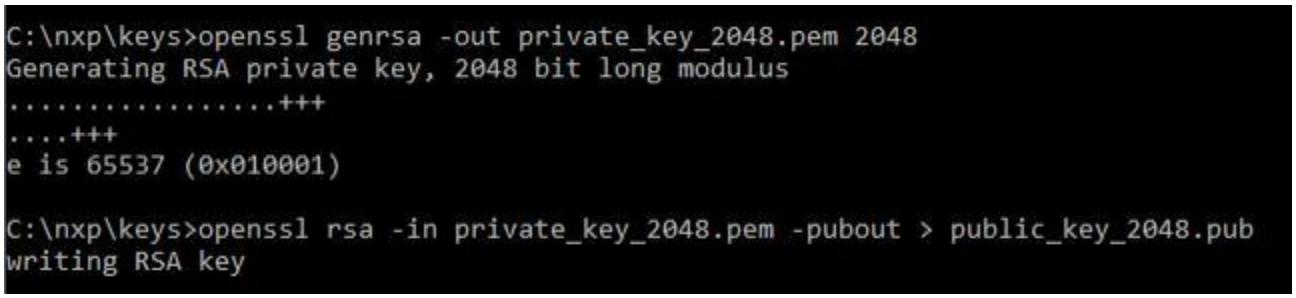

图5. 通过 OpenSSL 生成密钥对

或者，使用 NXPKEYGEN 工具在您的 VENV 中生成密钥对：

```
nxpkeygen -p 1.0 genkey keys\rot0_2048.pem
```

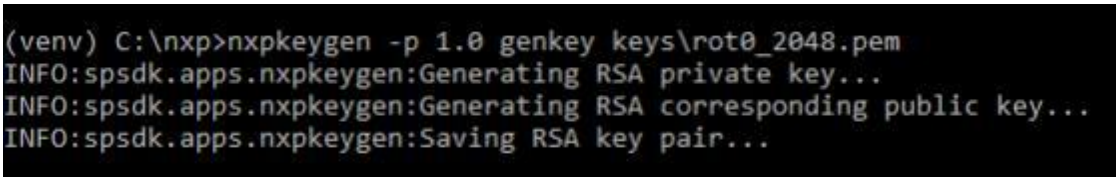

图6. 通过 NXPKEYGEN 生成密钥对

需要有四个可信任的根（RoT）密钥对和一个调试凭证密钥（DCK）对。

3.2.3 为设备创建配置

LPC55 系列器件具有称为受保护闪存区（PFR）的特殊存储器，此区域用于在板上或者以后更新期间进行设备配置。PFR 包括客户制造可编程区域（CMPA）和客户现场可编程区域（CFPA）。

最初，有可能生成一个示例配置文件：

```
pfr get-cfg-template -d lpc55s6x -t cmpa -o cmpa_config.yml
pfr get-cfg-template -d lpc55s6x -t cfpa -o cfpa_config.yml
```

对于调试鉴权，以下是最重要的寄存器：

CMPA:

CC_SOCU_PIN：用于配置并指定每个调试区域的调试访问权限。在PIN中，您可以设置 0 - 由调试身份鉴权控制的权限或 1- 始终固定的访问权限。

CC_SOCU_DFLT：用于配置并指定每个调试区域的调试访问权限。如果选择了PIN中的固定访问权限，则1-启用和 0-禁用。如果在 PIN 中选择了调试身份鉴权，则 DFLT 的唯一正确设置是 0。

PIN/DFLT 位域设置：

- 1/1 - 调试权限始终启用（固定）
- 1/0 - 调试权限始终禁用（固定）
- 0/0 - 使用调试身份验证启用调试权限

CC_SOCU_PIN 和 CC_SOCU_DFLT 寄存器中的以下位域为 MCU 配置不同的安全设置：

NIDEN：控制 CPU0 的 TrustZone 非安全域的非侵入式调试

DBGEN：控制 CPU0 的 TrustZone 非安全域的侵入式调试

SPNIDEN：控制 CPU0 的 TrustZone 安全域的非侵入式调试

SPIDEN：控制 CPU0 的 TrustZone 安全域的侵入式调试

TAPEN：控制 TAP（测试接入点）控制器

CPU1_DBGEN：控制 CPU1 的侵入式调试

ISP_CMD_EN：控制是否可以发出 ISP 启动流程命令

FA_CMD_EN：控制是否可以发出 Set FA Mode 命令

ME_CMD_EN：控制是否可以发出批量擦除命令

CPU1_NIDEN：控制 CPU1 的非侵入式调试

UUID_CHECK：在 DC 中 DAR 值指定的 UUID 期间启用检查

VENDOR_USAGE：在调试身份鉴权响应（DAR）处理期间，设备会检查调试证书的“Vendor Usage”字段中指定的值是否与设备配置的“VENDOR_USAGE”字段中编程的值完全匹配。其中高两个字节为CMPA.VENDOR_USAGE，低两个字节CFPA.VENDOR_USAGE。

RKTH：MCU 最多支持四个 RoT 密钥，用于安全启动和调试身份鉴权。稍后，可以撤销这些单独的 RoT 密钥。根密钥表哈希值（RKTH）是在 CMPA 中编程的值，它是 RoT 公钥的 SHA-256 哈希值。

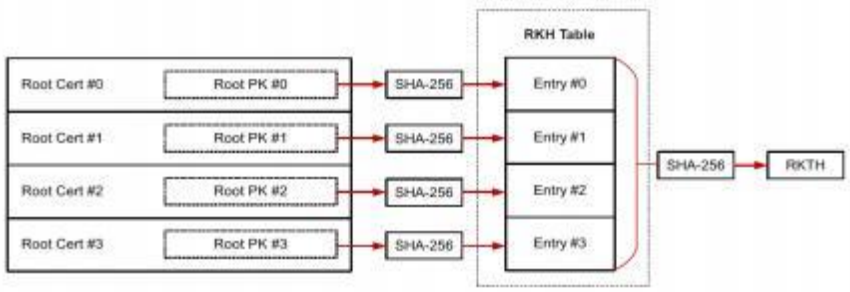

图7. RKTH生成过程

CFPA:

VERSION：CFPA 更新机制在每次更新时检查版本，新版本必须高于以前的版本。

ROTKH_REVOKE：在 MCU 中编程的 RoT 密钥可以在以后撤销。此字段指定哪些密钥已被撤销，哪些仍可用于调试身份鉴权和安全启动。

VENDOR_USAGE：在调试身份验证响应（DAR）处理期间，设备会检查调试证书的“Vendor Usage”字段中指定的值是否与设备配置的“VENDOR_USAGE”字段中编程的值完全匹配。其中高两个字节为CMPA.VENDOR_USAGE，低两个字节为CFPA.VENDOR_USAGE。

DCFG_CC_SOCU_PIN/DCFG_CC_SOCU_DFLT：这些可以进一步限制为 CMPA 中配置的调试权限。由于 CMPA 是在制造时编程的，因此可以稍后更新 CFPA 设置以收紧限制。CFPA 设置可以编程为与 CMPA 相同，以保持相同的限制。以下是 CFPA 与 CMPA 的不同组合，收紧了调试访问限制：

CMPA.DCFG_CC_SOCU -> CFPA.DCFG_CC_SOCU

- 固定启用 (1/1) -> 通过调试身份验证启用 (0/0)
- 固定启用 (1/1) -> 固定禁用 (1/0)
- 通过调试身份验证 (0/0) 启用 -> 固定禁用 (1/0)

配置JSON文件后，就可以生成二进制文件了。“pfr generate”命令的输入是 CMPA/CFPA JSON 配置输入文件 -c，在需要的地方计算反向寄存器 -i，可信任密钥根 -f（拥有正确的 RoT 密钥顺序很重要）或 -e 。如果使用 ELFTOSB 配置文件，BIN 文件的输出名称 -o 。例如，附加在 LPC55S69 设备的 cmpa_example_config.json 和 cfpa_example_config.json 文件（请记住，对于类似的 CFPA 版本有更新规则，它必须高于之前的 CFPA 内容等）。对于 cmpa.bin，仅使用这些选项之一。

```
pfr generate-binary -c cmpa_config.yml -i -e config_elftosb.json -o cmpa.bin
pfr generate-binary -c cmpa_config.yml -i -f keys\rotk0_rsa_2048.pub -f keys\rotk1_rsa_2048.pub -f
keys\rotk2_rsa_2048.pub -f keys\rotk3_rsa_2048.pub -o cmpa.bin
pfr generate-binary -c cfpa_config.json -i -o cfpa.bin
```

3.2.4 创建调试身份鉴权证书

NXP 提供了调试鉴权协议，这是一种质询-响应机制。当调试器通过调试邮箱向设备发送调试身份鉴权启动命令时，设备会生成调试身份验证质询（DAC）。DAC 消息包括以下元素，在 MCU 用户手册的“调试身份验证质询”部分中有进一步的说明。

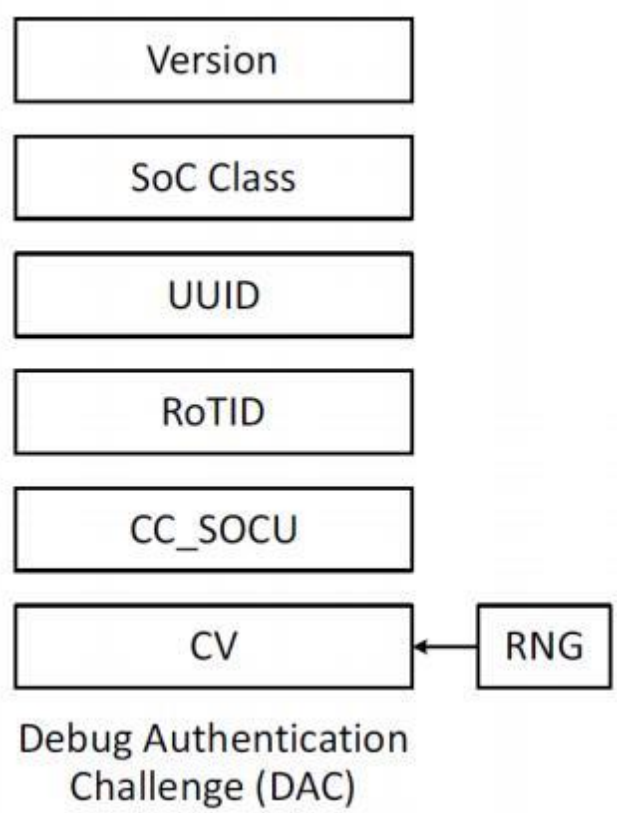

图8. 调试身份验证质询（DAC）字段

当调试身份验证工具收到此 DAC 时，它会计算调试身份验证响应（DAR），这里面包含了调试证书（DC）。

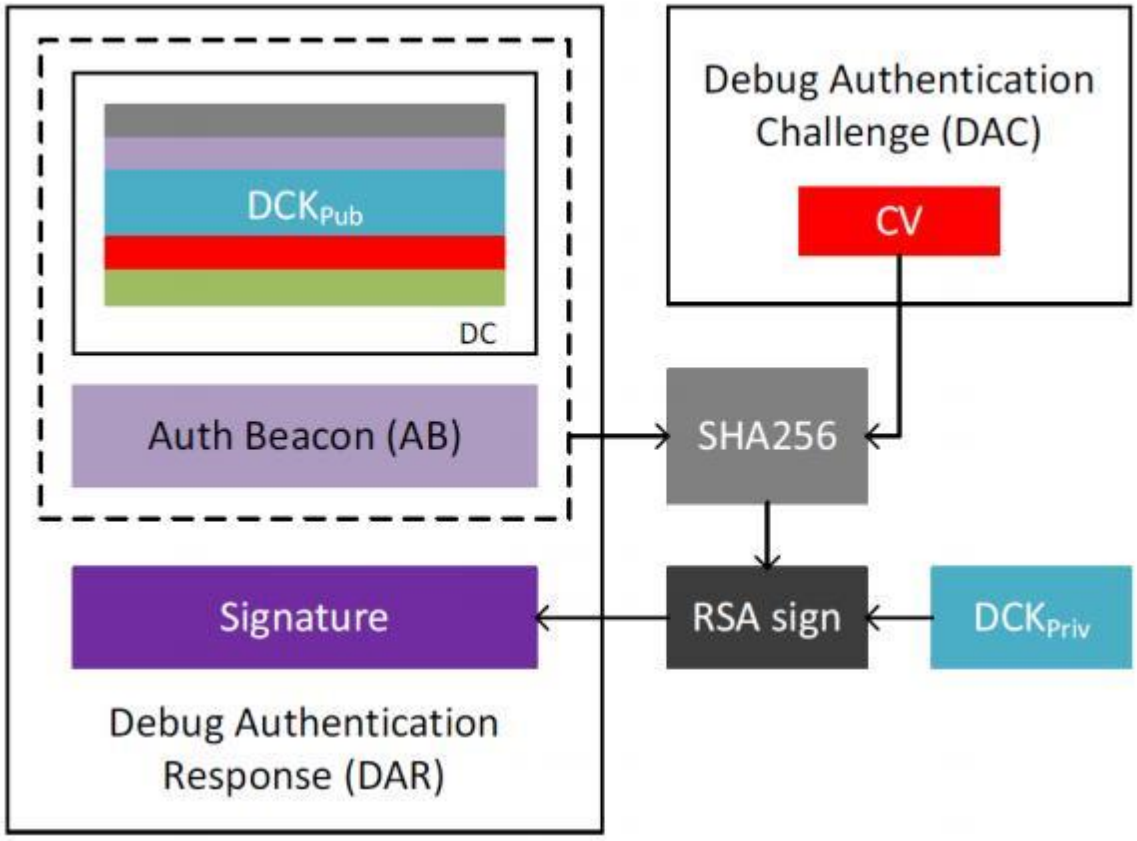

图9. 调试身份验证响应（DAR）字段

调试用户创建密钥对向 DC 进行身份验证。他们将 DC 公钥发送给 RoT 密钥的所有者，后者生成具有所有所需配置的 DC 证书（调试访问限制侵入性/非侵入性/安全/非安全、vendor_usage、凭据信标）。此证书利用 RoT 私钥签名进行身份验证。图 10展示了这个 DC 的结构。此示例命令用于为 DC 生成密钥对。

```
nxpkeygen -p 1.0 genkey keys\dck_rsa_2048.pem
```

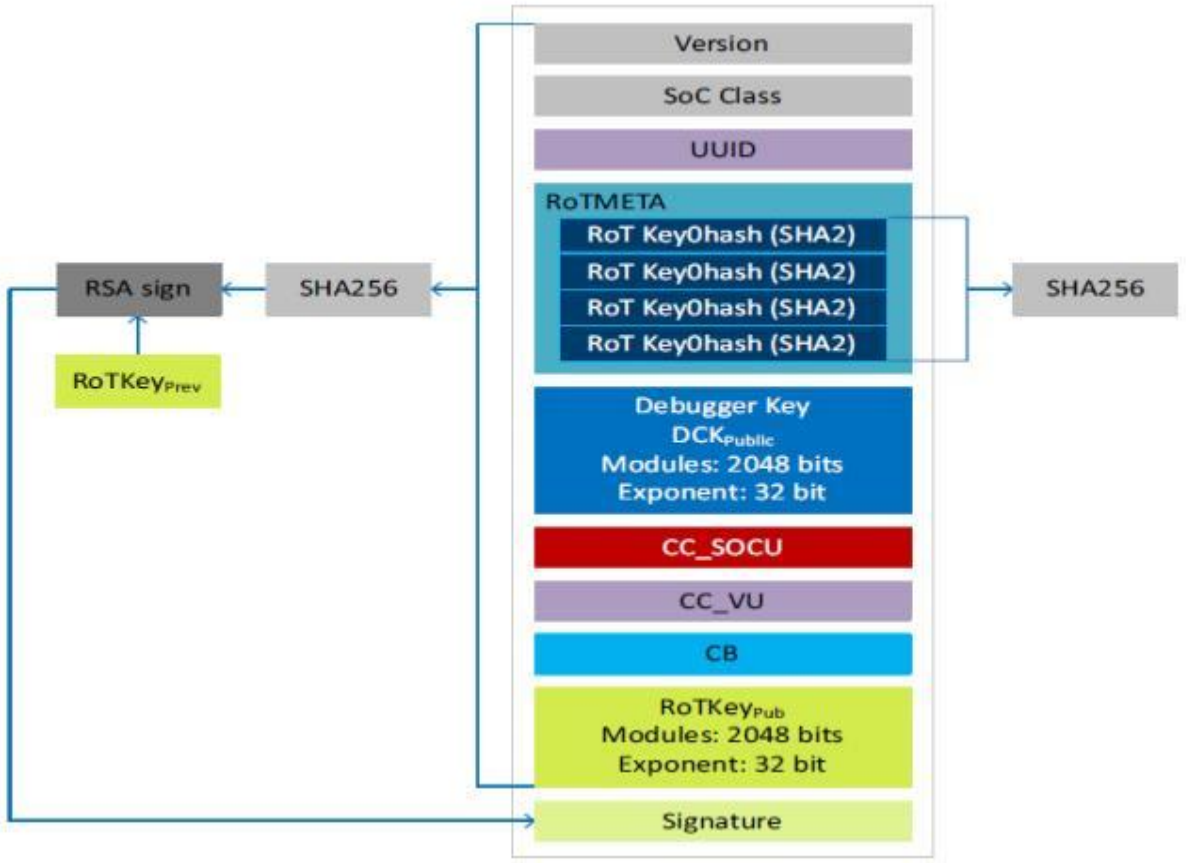

图10. 调试凭据证书（DC）字段

RoT 密钥所有者必须在 config.yaml 文件中配置调试访问限制。带有 RoT 路径的 ELFTOSB 配置文件可以与参数 -e 一起使用。在这种情况下，NXPKEYGEN 的配置文件中不需要“rot_meta”和“rotk”。以下是生成 DC 的 GENDC 命令的示例，仅使用以下选项之一：

```
nxpkeygen gendc -c keys\config.yml keys\dck_rsa_2048.dc

nxpkeygen gendc -c keys\config.yml -e keys\config_elftosb.json keys\dck_rsa_2048.dc
```

```
(venv) C:\nxp>nxpkeygen gendc -c keys\config.yml keys\dck_rsa_2048.dc
INFO:spsdk.apps.nxpkeygen:Loading configuration from yml file...
INFO:spsdk.apps.nxpkeygen:Creating RSA debug credential object...
INFO:spsdk.apps.nxpkeygen:Saving the debug credential to a file...
```

图11. 通过 NXPKEYGEN 生成 DC

在此之后，DC 文件由 NXPKEYGEN 工具生成，将来可以使用此调试证书，通过调试器进行身份验证。

3.2.5 将配置加载到设备中

需要在上一章中生成正确的配置，然后将此二进制配置加载到设备中。

默认情况下，所有设备都启用了调试探针，因此可以使用 IDE 加载和调试项目。例如，使用 MCUXpresso IDE。您可以调试并连接到设备。下面，使用“led_blinky”SDK示例。

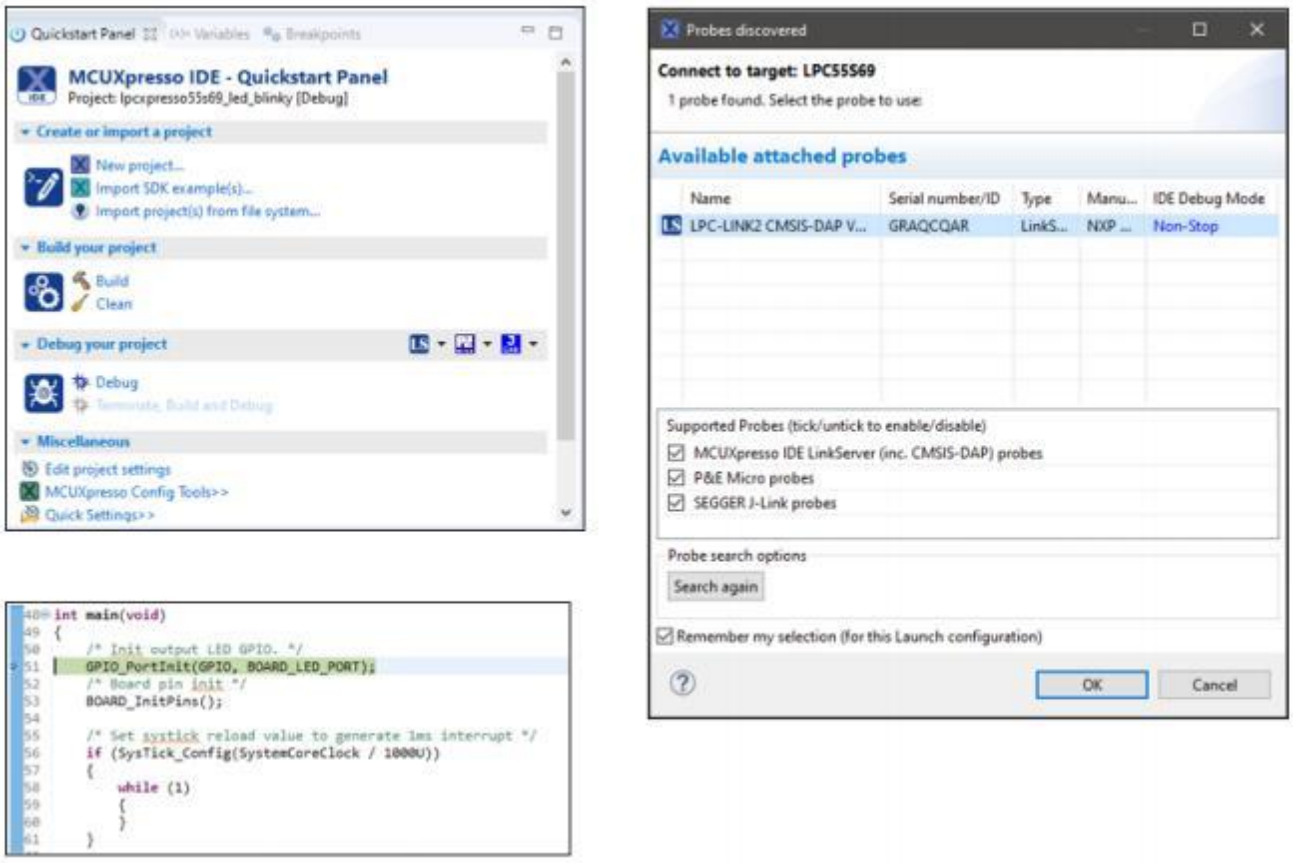

图12. 来自 MCUXpresso 的调试器连接

“led_blinky”示例项目已加载到板中。复位后，您可以看到 LED 闪烁。

对于加载设备配置，建议使用ISP模式，通过UART配置。

可以使用 BLHOST，它是 SPSDK 的一部分。对于 ISP 通信，设备必须进入 ISP 模式（按住 ISP 按钮并使用复位引脚对设备进行复位）并测试 ISP 通信。

```
blhost -p COMx get-property 1
```

此命令确认 BLHOST 正在与启动 ROM 通信，并且它应该收到来自设备的以下响应。

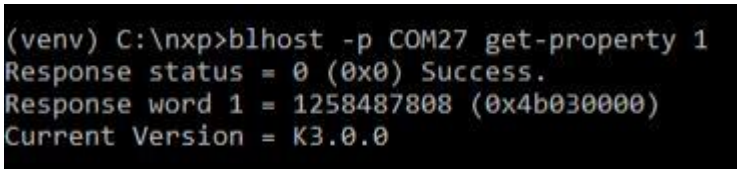

图13. 设备响应

测试过 ISP 通信后，可以将配置加载到设备中。预计在前面的步骤中已经使用正确的配置创建了 *cmpa.bin* 和 *cfpa.bin* 文件。

将 CMPA/CFPA 配置加载到 LPC55S6x/LPC55S2x 的示例如下所示。该器件的 PFR 起始地址为 0x9de00，LPC55S1x/LPC55S0x的PFR地址为0x3de00。

```
blhost -p COMx write-memory 0x9e400 cmpa.bin

blhost -p COMx write-memory 0x9de00 cfpa.bin
```

当您通过上电复位或引脚复位对设备进行复位时，会加载 *cmpa.bin* 和 *cfpa.bin* 文件中指定的配置。为调试探针启用 DAP 时，您将看到 IDE 在复位后找不到可用于调试连接的 SWD 设备。

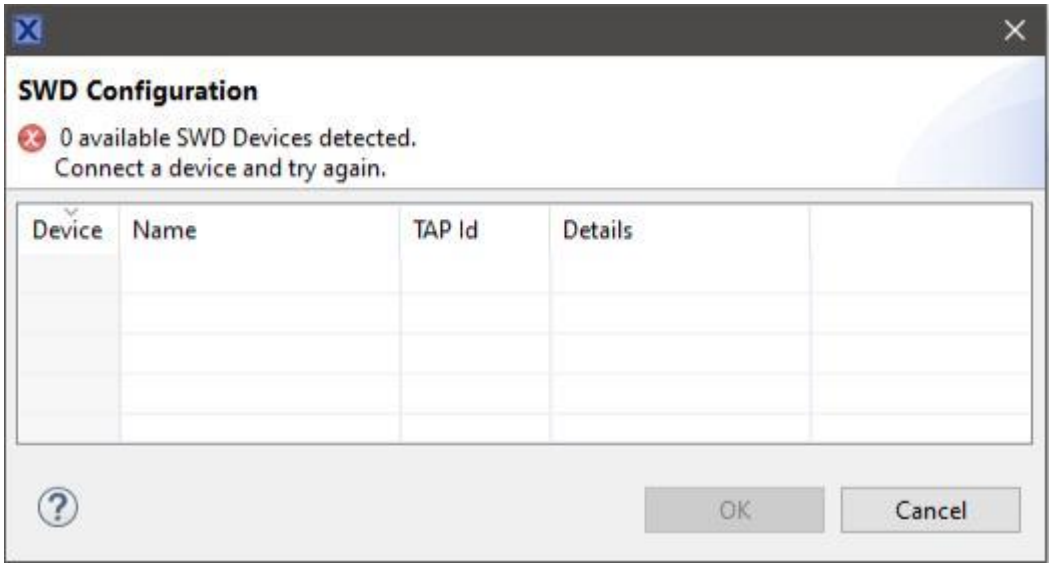

图14. 不可用的调试端口

3.2.6 使用调试认证工具打开调试端口

在此步骤中，我们可以测试一下作为SPSDK一部分的调试身份鉴权工具。此工具通过调试器探针与调试邮箱通信，并尝试进行身份验证以启用调试权限。接口选择 -i 可以支持用于 CMSIS-DAP 的 PYOCD 和用于 J-Link 调试器的 JLINK，来自该工具的一些进程信息会显示在控制台中。当调试验证成功时，就可以再次将调试器连接到核。例如，带有 MCUXpresso IDE 的“led_blinky”项目。-b 参数是一个认证信标，它通过 ROM 与“cc_beacon”一起从 DC 文件处理到“SYSCON->DEBUG_AUTH_BEACON”寄存器中，并且可以被在设备中运行的应用程序读取。-c 参数是 DC 文件，由 RoT 密钥的所有者签名，并经过身份鉴权以打开调试端口，如 DC 文件中配置的那样。-k 参数是在 DC 文件签名之前最初生成的私钥。它是保障 OEM（RoT 密钥所有者）和现场技术人员调试证书安全的方法。

```
nxpdebugmbox -p 1.0 auth -b 0 -c keys\dck_rsa_2048.dc -k keys\dck_rsa_2048.pem
```

调试鉴权通过后，设备上的调试端口被解锁，直到按下上电复位或引脚复位。调试器可以在不失去调试访问权限的情况下进行软件复位。

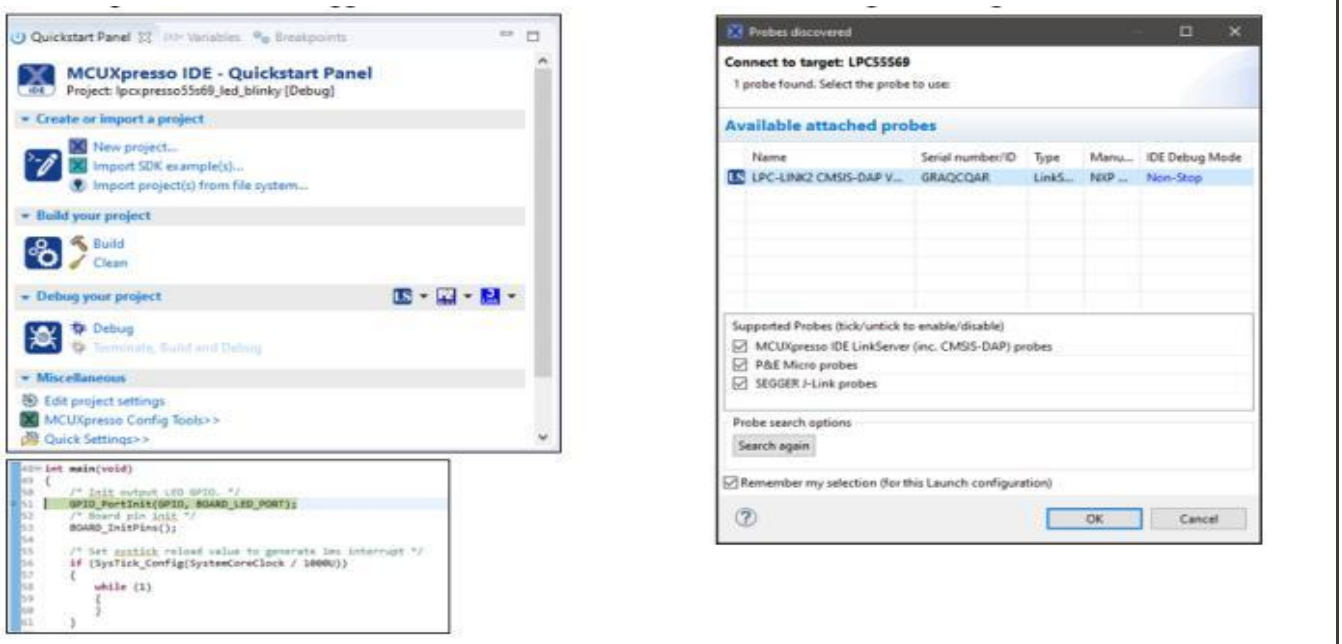

图15. 调试认证后可用的调试端口

4 总结

NXP 调试身份鉴权是在产品的整个生命周期内确保客户安全的关键功能之一。在本应用笔记中，描述了 LPC55S69 器件的示例配置和用法，其他 LPC55xx 器件的原理和工具与此类似。

5 参考资料

1. *LPC55S6x/LPC55S2x/LPC552x User Manual* (document UM11126)
2. *LPC55S6x Data Sheet* (document LPC55S6x)
3. *MCUXpresso IDE User's Guide*

6 修订历史

表1. 修订历史

修订号	日期	内容变化
1	2021年10月6日	SPSDK版本升级
0	2020年11月	初版发布

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: 06 October 2021

Document identifier: AN13037