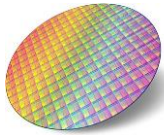


NFC Type 5 / RFID 标签 IC（包含至多 2.5 Kb EEPROM），产品识别及保护



UDFPN5

1.7 x 1.4 mm



晶圆



产品状态链接

[ST25TV02KC](#)

[ST25TV512C](#)

特征

非接触接口

- 符合 ISO/IEC 15693 标准
- 经 NFC Forum 认证的 NFC Forum Type 5 标签
- 支持所有 ISO/IEC 15693 调制、编码、副载波模式和高达 26 Kbit/s 的数据速率
- 单块读写，多块读取
- 内部调谐电容：23 pF

存储器

- 高达 2560 位（320 字节）的 EEPROM
- 可由四字节块访问
- RF 写入时间：1 个块通常 5 ms
- 数据保存期：55°C 下 60 年
- 最小耐久性：100k 写循环
- 3 位数的唯一输入码
- 增强型 NDEF（内容自动关联 NDEF 消息）

数据保护

- 用户存储器可在一个或两个区域中配置：
 - 在单区域模式下，可通过一个 64 位密码保护访问
 - 在灵活的双区域模式下，可通过两个 32 位密码保护访问
- 系统配置：通过 32 位密码保护访问
- 特定用户区块永久写入锁定
- 用户区域级别临时写入锁定
- 特定系统配置块永久写入锁定

产品识别及保护

- 密码功能：封面编码、恢复、失败尝试计数器
- 篡改检测功能可存储打开/重新密封的事件
- TruST25™ 数字签名

私有

- 可配置终止模式永久停用标签
- 具有可配置响应功能的不可追踪模式

温度范围

- 从 -40 至 85 °C

封装

- 5 引脚封装，ECOPACK2®（符合 RoHS 标准）
- 凸圆片和锯圆片

1 说明

ST25TV02KC 和 ST25TV512C 器件为 NFC / RFID 标签 IC，具有“增强型 NDEF”功能、篡改检测接口以及保护客户隐私的特定模式。

“增强型 NDEF”功能是一种内容自动关联 NDEF 消息服务，允许标签响应动态内容，无需最终用户显式更新 EEPROM。

篡改检测接口仅在 ST25TV02KC-T 器件上可用。该接口在 ST25TV02KC-A 和 ST25TV512C 器件上不可用。

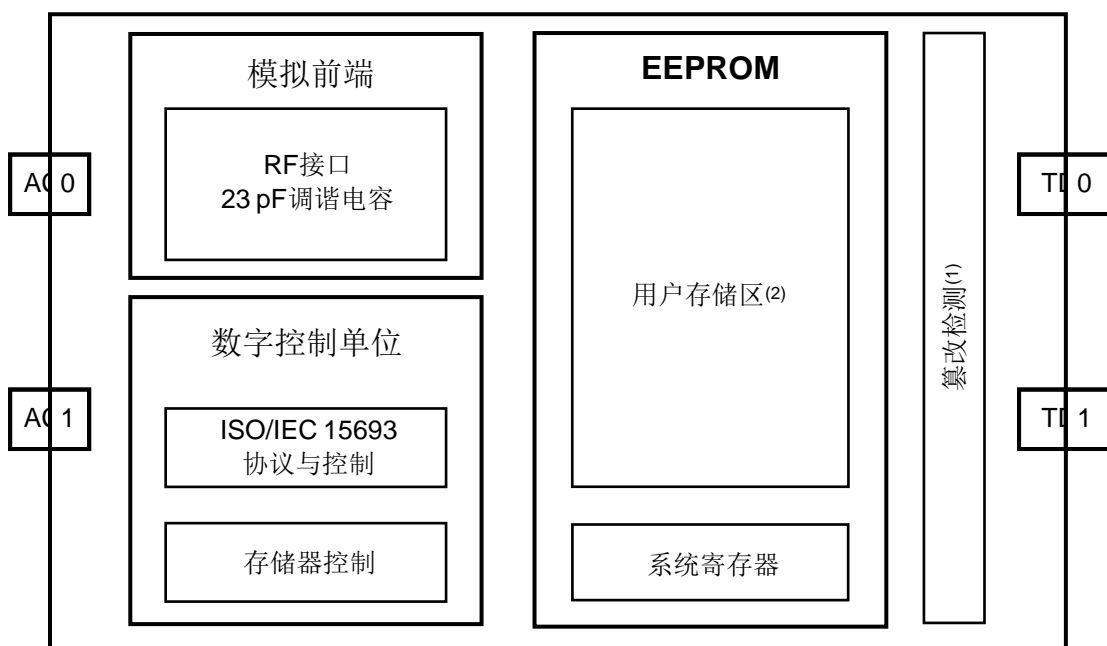
ST25TV02KC 和 ST25TV512C 器件具有 TruST25™ 生成的数字签名（一套软件和程序），以证明芯片在克隆检测中的来源，嵌入具有 60 年数据保存期的可配置 EEPROM，且可从 13.56 MHz 远程 RFID 读卡器或 NFC 手机进行操作。

非接触式接口符合 ISO/IEC 15693 标准和 NFC Forum Type 5 标签。

1.1 框图

ST25TV02KC 和 ST25TV512C（以下简称 ST25TVxxxC）器件如以下框图所述：

图 1. ST25TVxxxC 框图



1. 篡改检测接口仅在ST25TV02KC-T器件上可用。

2. 在ST25TV512C和ST25TV02KC器件上分别为512和2560位。

1.2 封装连接

ST25TV02KC ST25TV512C 以两种交付形式提供：

- UDFPN5 封装（仅限 ST25TV02KC-T 器件）
- 已切割植球晶圆（ST25TV512C、ST25TV02KC-A 和 ST25TV02KC-T 器件）

表 1. 信号名称

| 信号名称 | 功能 | 方向 |
|------|--------|-------|
| AC0 | 天线线圈 | 输入-输出 |
| AC1 | 天线线圈 | 输入-输出 |
| TD0 | 篡改检测回路 | 输入 |
| TD1 | 篡改检测回路 | 输出 |

图 2. UFDFPN5 封装连接

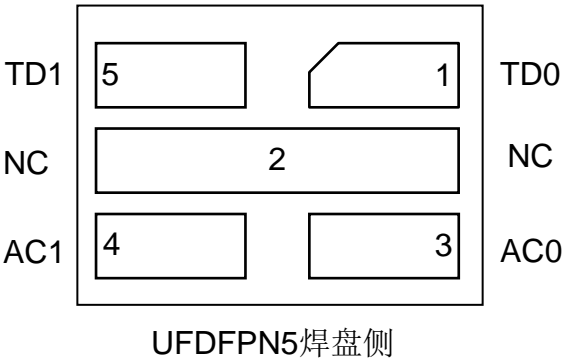
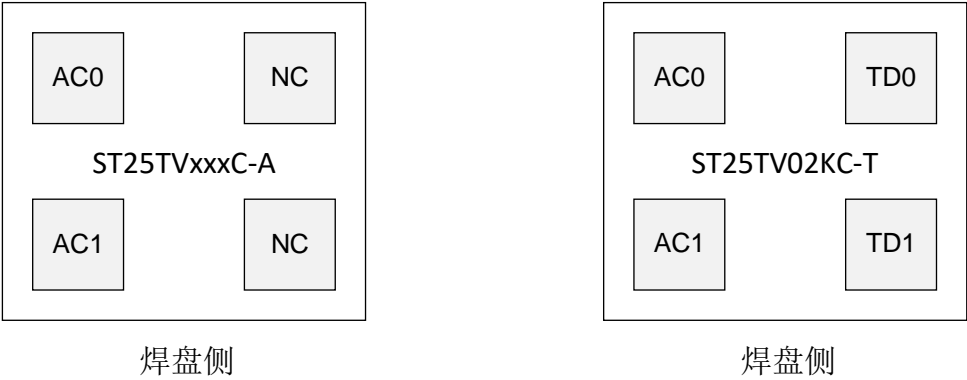


图 3. 已切割植球晶圆的晶片连接



2 信号描述

2.1 天线线圈（AC0，AC1）

这些输入用于专门将 ST25TVxxxC 设备连接到外部线圈。建议不要将其它任何 DC 或 AC 路径连接到 AC0 或 AC1。

当调谐正确时，该线圈用于使用 ISO/IEC 15693 和 ISO 18000-3 模式 1 协议供电和访问设备。

2.2 篡改检测（TD0，TD1）

这些输入用于将一个线环连接到 ST25TVxxxC 器件，以检测 TD0 和 TD1 引脚之间的开路或短路。

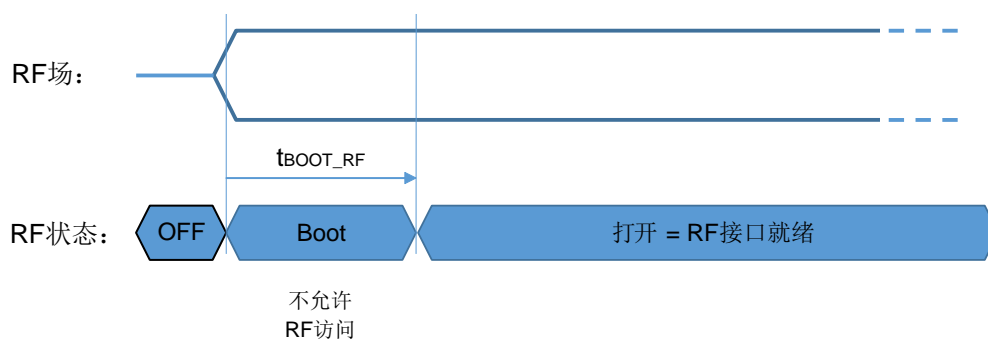
3 电源管理

ST25TVxxxC 器件通过非接触接口供电。

3.1 器件设置

为确保 RF 电路正确复位，必须将 RF 场至少开启（无任何调制）一段时间 t_{BOOT_RF} （参见表 173. RF 特性）。在 t_{BOOT_RF} 期间，ST25TVxxxC 忽略接收到的全部 RF 指令（参见图 4）。

图 4. RF 上电时序



3.2 器件复位

为确保 RF 电路正确复位，必须将 RF 场（100%调制）关闭至少一段时间 t_{RF_OFF} （参见表 173. RF 特性）。

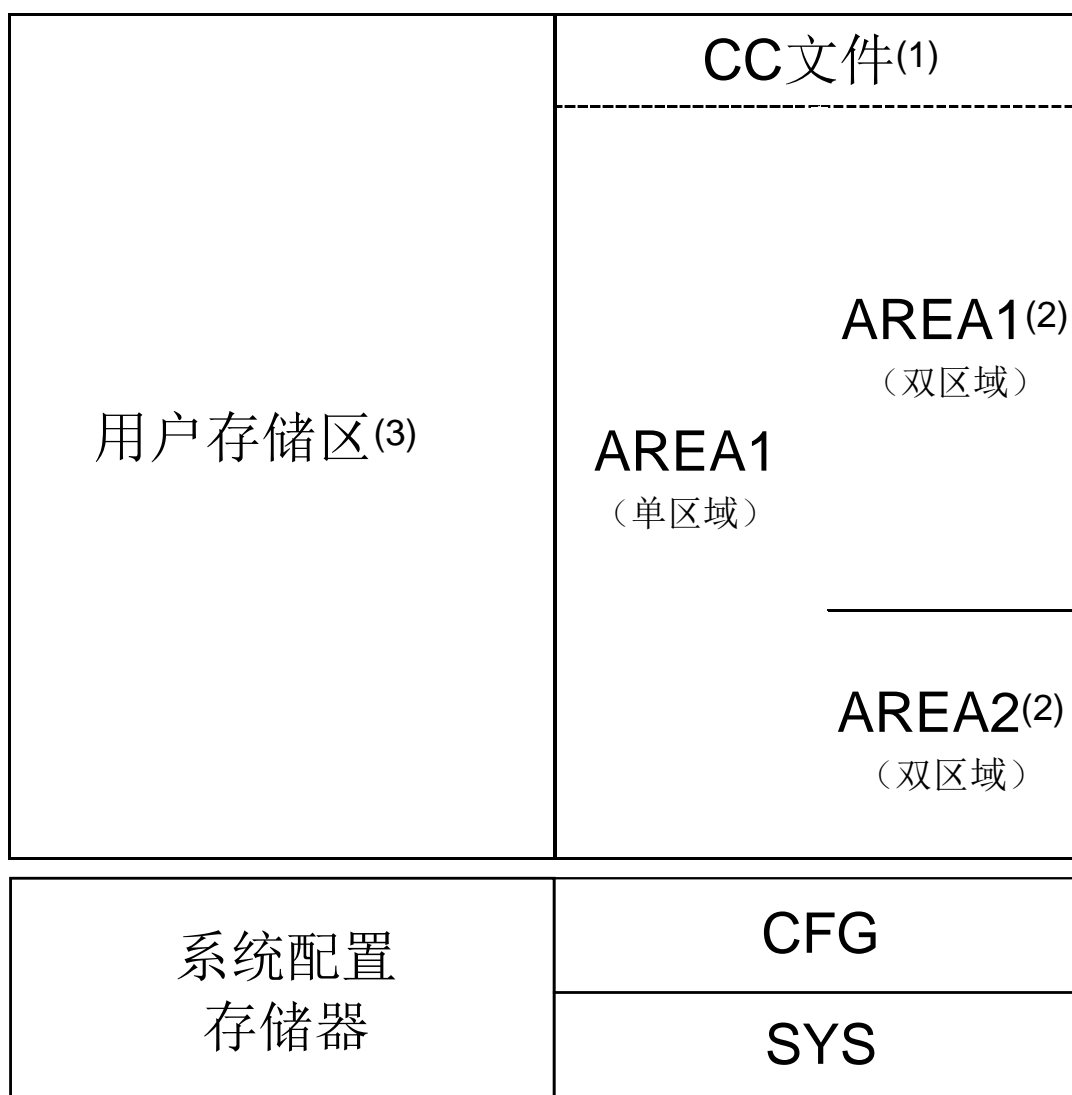
4 存储器管理

4.1 存储器组织结构

ST25TVxxxC 存储器组织结构如下：

- 用户存储器：可在一个或两个不同的区域中配置，如第 4.2 节 用户存储器中所述。如果需要，这些区域可用作用户数据和 NFC Forum Type 5 性能容器（CC）。
- 系统配置存储器：它由不同的寄存器组成，包括器件配置寄存器 ISO15693 AFI 和 DSFID。它还包含 UID 和不同的保护寄存器。更多详细信息，请参见第 4.3 节 系统配置存储器。

图 5. 存储器组织结构



1. NFC Forum T5T CC 文件在块 00h 上进行编码，此块为 AREA1 的一部分。
2. 在双区域模式中，可对 AREA1/AREA2 边界进行块间隔尺寸配置。
3. ST25TV512C 和 ST25TV02KC 器件分别有 16 个和 80 个 32 位的块。

4.2 用户存储器

用户存储器以 4 字节的块 (= 页面) 寻址, 这些块以地址 0 开始, 并以地址 END_MEM 结束。ST25TV512C 和 ST25TV02KC 器件的 END_MEM 值分别为 0Fh 和 4Fh。根据 RF 会话开始时的 END_A1 寄存器的值, 可在单区域 (AREA1) 或双区域 (AREA1 和 AREA2) 模式中配置 ST25TVxxxC 用户存储器 (参见表 2. 单区域模式中的用户存储器、表 3. 双区域模式中的用户存储器和表 13. END_A1 内容)。

当 END_A1 的值等于 END_MEM 时, ST25TVxxxC 用户存储器被配置成如下的单区域模式:

- AREA1 从地址 00h 开始。它由 (END_MEM+1) 块组成。可以通过专用的 64 位密码实现读或写保护。AREA1 专门用于用户数据。

当 END_A1 的值小于 END_MEM 时, ST25TVxxxC 用户存储器用户存储器被配置成如下的双区域模式:

- AREA1 从地址 00h 开始。它由 (END_A1+1) 块组成。可以通过专用的 32 位密码实现读或写保护。AREA1 专门用于用户数据。
- AREA2 从地址 (END_A1+1) 开始。它由 (END_MEM-END_A1) 块组成。可以通过专用的 32 位密码实现读或写保护。AREA2 专门用于用户数据。

块 00h 属于 AREA1, 但无论 AREA1 的读保护模式如何, 始终可以读取。此块专用于 NFC Forum Type 5 应用定义的 CC 文件内容。应用无需符合 NFC Forum Type 5 规范即可将块 00h 用于任何目的。

表 2. 单区域模式中的用户存储器

| RF 指令 | 块地址 | 字节地址 | | | | 备注 |
|---|--------------------|-------------|-------------|-------------|-------------|----------------------|
| | | LSByte | - | - | MSByte | |
| ReadSingleBlock ReadMultipleBlocks WriteSingleBlock | 00h ⁽¹⁾ | 0000h | 0001h | 0002h | 0003h | AREA1 ⁽²⁾ |
| | 01h | 0004h | 0005h | 0006h | 0007h | |
| | 02h | 0008h | 0009h | 000Ah | 000Bh | |
| | ... | ... | ... | ... | ... | |
| | END_MEM | END_MEM*4+0 | END_MEM*4+1 | END_MEM*4+2 | END_MEM*4+3 | |

- 块 00h 始终可读
- 对于单区域模式, 将 END_A1 寄存器的值设为 END_MEM

表 3. 双区域模式中的用户存储器

| RF 指令 | 块地址 | 字节地址 | | | | 备注 |
|---|--------------------|-------------|-------------|-------------|-------------|----------------------|
| | | LSByte | - | - | MSByte | |
| ReadSingleBlock ReadMultipleBlocks WriteSingleBlock | 00h ⁽¹⁾ | 0000h | 0001h | 0002h | 0003h | AREA1 ⁽²⁾ |
| | 01h | 0004h | 0005h | 0006h | 0007h | |
| | 02h | 0008h | 0009h | 000Ah | 000Bh | |
| | ... | ... | ... | ... | ... | |
| | END_A1 | END_A1*4+0 | END_A1*4+1 | END_A1*4+2 | END_A1*4+3 | |
| | END_A1+1 | END_A1*4+4 | END_A1*4+5 | END_A1*4+6 | END_A1*4+7 | AREA2 ⁽²⁾ |
| | ... | ... | ... | ... | ... | |
| | END_MEM | END_MEM*4+0 | END_MEM*4+1 | END_MEM*4+2 | END_MEM*4+3 | |

- 块 00h 始终可读
- 对于双区域模式, 将 END_A1 寄存器的值设在 00h 与 (END_MEM-1) 之间。

4.3 系统配置存储器

除了用户存储器外, ST25TVxxxC 还包括一组位于系统配置存储器中的寄存器。寄存器在启动序列期间读取, 并定义了基本 ST25TVxxxC 行为。

4.3.1 系统配置寄存器

表 4. 配置寄存器列表 列出了 ST25TVxxxC 器件的配置寄存器。可通过 ReadConfiguration 与 WriteConfiguration 指令以及分别用作特征标识符与参数标识符的两个参数 FID 和 PID 来访问它们。

对配置寄存器的写访问受 CONFIG 安全会话保护，成功输入 PWD_CFG 密码后，可打开该安全会话（参见第 5.1.2 节 密码管理）。

按 FID 值对配置寄存器进行分组。可永久锁定对一组寄存器的写访问。

根据配置寄存器的不同，始终向其授予读访问权限，或对其读访问受到与写访问相同的机制保护。

根据配置寄存器的不同，当在 RF 会话期间更新其内容时，新值的影响立即生效或在下一个 RF 会话开始时生效。

表 4. 配置寄存器列表

| 名称 | FID | PID | 字节 | 读 ⁽¹⁾ | 写 ⁽²⁾ | 激活时间 ⁽³⁾ | 章节 |
|-----------------------------------|-----|-----|----|------------------|------------------|---------------------|-----------|
| RW_PROTECTION_A1 | 00h | 00h | 1 | Y | W | B | 第 5.1.1 节 |
| END_A1 | 00h | 01h | 1 | Y | W' | B | 第 5.1.1 节 |
| RW_PROTECTION_A2 | 01h | 00h | 1 | Y | W | B | 第 5.1.1 节 |
| UTC_EN | 02h | 00h | 1 | Y | W | B | 第 5.2.1 节 |
| UTC | 02h | 01h | 3 | Y | N | - | 第 5.2.1 节 |
| TD_EVENT_UPDATE_EN ⁽⁴⁾ | 03h | 00h | 1 | Y | W | B | 第 5.3.1 节 |
| TD_SEAL_MSG ⁽⁴⁾ | 03h | 01h | 2 | R | W | I | 第 5.3.1 节 |
| TD_UNSEAL_MSG ⁽⁴⁾ | 03h | 02h | 2 | R | W | I | 第 5.3.1 节 |
| TD_RESEAL_MSG ⁽⁴⁾ | 03h | 03h | 2 | R | W | I | 第 5.3.1 节 |
| TD_SHORT_MSG ⁽⁴⁾ | 03h | 04h | 1 | R | W | I | 第 5.3.1 节 |
| TD_OPEN_MSG ⁽⁴⁾ | 03h | 05h | 1 | R | W | I | 第 5.3.1 节 |
| TD_STATUS ⁽⁴⁾ | 03h | 06h | 3 | Y | N | - | 第 5.3.1 节 |
| ANDEF_EN | 04h | 00h | 1 | Y | W | B | 第 5.4.1 节 |
| ANDEF_CFG | 04h | 01h | 2 | Y | W | B | 第 5.4.1 节 |
| ANDEF_SEP | 04h | 02h | 1 | R | W | I | 第 5.4.1 节 |
| ANDEF_CUSTOM_LSB | 04h | 03h | 4 | R | W | I | 第 5.4.1 节 |
| ANDEF_CUSTOM_MSB | 04h | 04h | 4 | R | W | I | 第 5.4.1 节 |
| PRIVACY | 05h | 00h | 1 | Y | W | B | 第 5.5.1 节 |
| AFI_PROT | 08h | 00h | 1 | Y | W | B | 第 5.7.1 节 |
| REV | FEh | 00h | 1 | Y | N | - | 第 5.9 节 |
| UID | FEh | 01h | 8 | Y | N | - | 第 5.9 节 |
| LCK_CONFIG | FFh | 00h | 2 | Y | W'' | I | 第 5.1.1 节 |

1. Y: 读访问不受保护，如果 LCK_CONFIG[FID]=0b 且 CONFIG 安全会话打开，则授予读访问权限
2. N: 写访问不可用，W: 如果 LCK_CONFIG[FID]=0b 且 CONFIG 安全会话打开，则授予写访问权限，W': 如果 LCK_CONFIG[1:0]=00b 且 CONFIG 安全会话打开，则授予写访问权限，W'': 如果 CONFIG 安全会话打开，则授予写访问权限
3. B: 更新在下一个 RF 启动序列生效，I: 更新立即生效
4. FID=03h 的寄存器仅在 ST25TV02KC-T 器件上可用（参见第 10 节）

4.3.2 系统寄存器

表 5 列出了 ST25TVxxxC 器件的系统寄存器。可使用除 ReadConfiguration 和 WriteConfiguration 以外的其他 RF 命令进行访问。

当可对系统寄存器进行写访问时，可通过密码和/或锁定机制提供保护。

当可对系统寄存器进行读访问时，始终通过相应的 RF 命令授予读访问权限。

在系统寄存器的内容更新后，新值的效果会被立即激活。

表 5. 系统寄存器列表

| 名称 | 字节 | 可读 ⁽¹⁾ | 可写 ⁽²⁾ | 激活时间 | 章节 |
|------------|----|-------------------|-------------------|------|---|
| LCK_BLOCK | 10 | R | W | I | 第 5.1.1 节 数据保护寄存器 |
| LCK_DSfid | 1 | N | W | I | 第 5.9 节 设备识别寄存器 |
| LCK_AFI | 1 | N | W | I | 第 5.9 节 设备识别寄存器 |
| DSfid | 1 | Y | W | I | 第 5.9 节 设备识别寄存器 |
| AFI | 1 | Y | W | I | 第 5.9 节 设备识别寄存器 |
| IC_REF | 1 | Y | N | - | 第 5.9 节 设备识别寄存器 |
| UID | 8 | Y | N | - | 第 5.9 节 设备识别寄存器 |
| ANDEF_UID | 16 | Y | N | - | 第 5.4.1 节 增强型 NDEF 寄存器 |
| KILL_CMD | 1 | N | W | I | 第 5.5.1 节 增强型 NDEF messagePrivacy 寄存器示例 |
| UNTR_CMD | 1 | N | W | I | 第 5.5.1 节 增强型 NDEF messagePrivacy 寄存器示例 |
| RND_NUMBER | 2 | Y | N ⁽³⁾ | - | 第 5.1.1 节 数据保护寄存器 |
| PWD_CFG | 4 | N | W | I | 第 5.1.1 节 数据保护寄存器 |
| PWD_A1 | 4 | N | W | I | 第 5.1.1 节 数据保护寄存器 |
| PWD_A2 | 4 | N | W | I | 第 5.1.1 节 数据保护寄存器 |
| PWD_UNTR | 4 | N | W | I | 第 5.1.1 节 数据保护寄存器 |

1. Y: 无条件地授予读访问权限, R: 有条件地授予读访问权限
2. N: 不允许写访问, W: 有条件地授予写访问权限
3. 成功执行 *GetRandomNumber* 请求后, 在内部更新 *RND_NUMBER* 寄存器的内容

5 特性

ST25TVxxxC 提供了下列功能：

- 第 5.1 节 数据保护
- 第 5.2 节 唯一输入码
- 第 5.3 节 篡改检测
- 第 5.4 节 增强型 NDEF
- 第 5.5 节 消费者隐私保护
- 第 5.6 节 TruST25™ 数字签名
- 第 5.7 节 AFI 保护
- 第 5.8 节 Inventory Initiated

通过使用 ReadConfiguration 与 WriteConfiguration 指令来访问 ST25TVxxxC 寄存器，可对从第 5.1 节到第 5.7 节的功能进行编程。仅在通过输入配置密码 (PWD_CFG) 授予访问权限后，并且功能配置之前没有锁定（参见 LCK_CONFIG 寄存器）时，才能更新配置寄存器。

根据配置寄存器的不同，可立即应用有效写访问的效果，或在下一个 RF 会话的启动序列期间应用。

一组额外的寄存器便于确定和自定义产品（参见第 5.9 节 设备识别寄存器）

5.1 数据保护

ST25TVxxxC 提供基于安全会话解锁密码的特殊数据保护机制。

对用户存储器的读和/或写访问可受保护。对配置寄存器的写访问始终受保护。对一些配置寄存器的读访问受保护。支持本节中所述的其他锁定机制（LockBlock、按功能锁定）。

5.1.1 数据保护寄存器

表 6. LCK_CONFIG 访问

| RF 指令 | 访问类型 |
|--|----------------------|
| ReadConfiguration @(FID=FFh, PID=00h) | R: 始终可能 |
| WriteConfiguration @(FID=FFh, PID=00h) | W: 如果 CONFIG 安全会话已打开 |
| | W 生效时间: 立即 |

表 7. LCK_CONFIG 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|-----------|--|----------|
| b0 | LCK_A1 | 0: FID=00h 的配置寄存器未锁定 1: FID=00h 的配置寄存器已锁定 | 0b |
| b1 | LCK_A2 | 0: FID=01h 的配置寄存器未锁定 1: FID=01h 的配置寄存器已锁定 | 0b |
| b2 | LCK_UTC | 0: FID=02h 的配置寄存器未锁定 1: FID=02h 的配置寄存器已锁定 | 0b |
| b3 | LCK_TD | 0: FID=03h 的配置寄存器未锁定 1: FID=03h 的配置寄存器已锁定 | 0b |
| b4 | LCK_ANDEF | 0: FID=04h 的配置寄存器未锁定 1: FID=04h 的配置寄存器已锁定 | 0b |
| b5 | LCK_PRIV | 0: FID=05h 的配置寄存器未锁定 1: FID=05h 的配置寄存器已锁定 | 0b |
| b7-b6 | RFU | - | 00b |
| b8 | LCK_AFIP | 0: FID=08h 的配置寄存器未锁定 1: FID=08h 的配置寄存器已锁定 | 0b |
| b15-b9 | RFU | - | 0000000b |

提示

有关 **LCK_CONFIG** 寄存器，请参见表 4. 配置寄存器列表。

如果向设为 1b 的字段发出值 1b，则 **WriteConfiguration** 命令没有效果，应以错误 11h 回应。

否则，如果向设为 1b 的字段发出值 0b，则相应的功能保持锁定，不向 **WriteConfiguration** 指令回应错误代码。

表 8. LCK_BLOCK 访问

| RF 指令 | 访问类型 |
|--------------------------------|---|
| ReadSingleBlock | R: 如果 Option_flag=1 且允许对父区域进行写访问 ⁽¹⁾ |
| ReadMultipleBlocks | R: 如果 Option_flag=1 且允许对父区域进行写访问 ⁽¹⁾ |
| GetMultipleBlockSecurityStatus | R: 如果允许对父区域进行写访问 ⁽¹⁾ |
| LockBlock | W: 如果尚未锁定并允许对父区域进行写访问 |
| - | W 生效时间: 立即 |

1. 当不允许对一个区域进行写访问时（写访问被禁止，或通过关闭的安全会话受到保护），则 **LCK_BLOCK** 的值被 **BSS** 字段中的 1 屏蔽（参见第 6.4.3 节、第 6.4.6 节和第 6.4.14 节）

表 9. LCK_BLOCK 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|-----------|--|-----|
| b79-b0 | LCK_BLOCK | 对于每个位 b_N : <ul style="list-style-type: none"> 0: 块 N 的写访问未锁定 1: 块 N 的永久写访问 | 0 |

提示

有关 **LCK_BLOCK** 寄存器，请参见表 5. 系统寄存器列表。

表 10. RW_PROTECTION_A1 访问

| RF 指令 | 类型 |
|--|---------------------------------|
| ReadConfiguration @(FID=00h, PID=00h) | R: 始终可能 |
| WriteConfiguration @(FID=00h, PID=00h) | W: 如果 CONFIG 安全会话已打开且 LCK_A1=0b |
| | W 生效时间: 在下一个 RF 启动序列 |

表 11. RW_PROTECTION_A1 内容

| 位 | 名称 | 功能 | 出厂值 |
|-------|------------------|---|---------|
| b1-b0 | RW_PROTECTION_A1 | <p>AREA1 访问权限（除了块 00h）：</p> <ul style="list-style-type: none"> 00: 始终允许读/写访问 01: 读始终允许/如果 AREA1 安全会话打开，则允许写 10: 如果 AREA1 安全会话打开，则允许读/如果 AREA1 安全会话打开，则允许写 11: 如果 AREA1 安全会话打开，则允许读/禁止写 <p>块 00h 访问权限：始终允许读</p> <ul style="list-style-type: none"> 00: 始终允许写 01: 如果 AREA1 安全会话打开，则允许写 10: 如果 AREA1 安全会话打开，则允许写 11: 禁止写 | 00b |
| b7-b2 | RFU | - | 000000b |

提示

有关 RW_PROTECTION_A1 寄存器，请参见表 4. 配置寄存器列表。

表 12. END_A1 访问

| RF 指令 | 访问类型 |
|--|--|
| ReadConfiguration @(FID=00h, PID=01h) | R: 始终可能 |
| WriteConfiguration @(FID=00h, PID=01h) | W: 如果 CONFIG 安全会话已打开且 LCK_A1=LCK_A2=0b |
| | W 生效时间: 在下一个 RF 启动序列 |

表 13. END_A1 内容

| 位 | 名称 | 功能 | 出厂值 |
|-------|--------|---|------------------------|
| b7-b0 | END_A1 | <p>属于 AREA1 的最后一个块的编号。</p> <p>当小于 END_MEM 时，用户存储器被分为两个区域：</p> <ul style="list-style-type: none"> AREA1（块 00h 至 END_A1） AREA2（块 END_A1 + 1 至 END_MEM）。 <p>否则，用户存储器只包含一个区域：AREA1（块 00h 至 END_MEM）</p> | END_MEM ⁽¹⁾ |

1. ST25TV512C / ST25TV02KC 器件的 END_MEM 值分别为 0Fh / 4Fh。

提示

有关 END_A1 寄存器，请参见表 4. 配置寄存器列表。

表 14. RW_PROTECTION_A2 访问

| RF 指令 | 访问类型 |
|--|---------------------------------|
| ReadConfiguration @(FID=01h, PID=00h) | R: 始终可能 |
| WriteConfiguration @(FID=01h, PID=00h) | W: 如果 CONFIG 安全会话已打开且 LCK_A2=0b |
| | W 生效时间: 在下一个 RF 启动序列 |

表 15. RW_PROTECTION_A2 内容

| 位 | 名称 | 功能 | 出厂值 |
|-------|------------------|---|---------|
| b1-b0 | RW_PROTECTION_A2 | AREA2 访问权限: <ul style="list-style-type: none"> 00: 始终允许读/写访问 01: 读始终允许/如果 AREA2 安全会话打开, 则允许写 10: 如果 AREA2 安全会话打开, 则允许读/如果 AREA2 安全会话打开, 则允许写 11: 如果 AREA2 安全会话打开, 则允许读/不允许写 | 00b |
| b7-b2 | RFU | - | 000000b |

提示

有关 RW_PROTECTION_A2 寄存器, 请参见表 4. 配置寄存器列表。

表 16. PWD_CFG 访问

| RF 指令 | 访问类型 |
|------------------------|----------------------|
| - | R: 无读访问 |
| WritePassword @PID=00h | W: 如果 CONFIG 安全会话已打开 |
| | W 生效时间: 立即 |

表 17. PWD_CFG 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|---------|---------------------|-----------|
| b31-b0 | PWD_CFG | 配置寄存器和 Kill 指令的访问密码 | 00000000h |

提示

有关 PWD_CFG 寄存器, 请参见表 5. 系统寄存器列表。

表 18. PWD_A1 访问

| RF 指令 | 访问类型 |
|------------------------|--------------------|
| - | R: 无读访问 |
| WritePassword @PID=01h | W: 如果 AREA1 安全会话打开 |
| | W 生效时间: 立即 |

表 19. PWD_A1 内容

| 位 | 名称 | 功能 | 出厂值 |
|---------|---------------------------|--|-----------|
| b31-b0 | PWD_A1_LSB | AREA1 的访问密码 | 00000000h |
| b63-b32 | PWD_A1_MSB ⁽¹⁾ | 在单区域模式中, 使用位 0 到 63。 在双区域模式中, 仅使用位 0 到 31 | 00000000h |

1. PWD_A1_MSB 为 PWD_A2 寄存器的别名:

- 当从双区域模式切换为单区域模式时, PWD_A1_MSB 的值为 PWD_A2 的最后已知值
- 当从单区域模式切换为双区域模式时, PWD_A2 的值为 PWD_A1_MSB 的最新已知值

提示

有关 PWD_A1 寄存器, 请参见表 5. 系统寄存器列表。

表 20. PWD_A2 访问

| RF 指令 | 访问类型 |
|------------------------|--------------------|
| - | R: 无读访问 |
| WritePassword @PID=02h | W: 如果 AREA2 安全会话打开 |
| | W 生效时间: 立即 |

表 21. PWD_A2 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|-----------------------|-------------|-----------|
| b31-b0 | PWD_A2 ⁽¹⁾ | AREA2 的访问密码 | 00000000h |

1. PWD_A1_MSB 为 PWD_A2 寄存器的别名:

- 当从双区域模式切换为单区域模式时, PWD_A1_MSB 的值为 PWD_A2 的最后已知值
- 当从单区域模式切换为双区域模式时, PWD_A2 的值为 PWD_A1_MSB 的最新已知值

提示

有关 PWD_A2 寄存器, 请参见表 5. 系统寄存器列表。

表 22. PWD_UNTR 访问

| RF 指令 | 访问类型 |
|------------------------|-------------------|
| - | R: 无读访问 |
| WritePassword @PID=03h | W: 如果 UNTR 安全会话打开 |
| | W 生效时间: 立即 |

表 23. PWD_UNTR 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|----------|----------------------------|-----------|
| b31-b0 | PWD_UNTR | 用于 ToggleUntraceable 指令的密码 | 00000000h |

提示

有关 PWD_UNTR 寄存器, 请参见表 5. 系统寄存器列表。

表 24. RND_NUMBER 访问

| RF 指令 | 访问类型 |
|-----------------|------------------------|
| GetRandomNumber | R: 始终可能 |
| - | W: 无写访问 ⁽¹⁾ |

1. 成功执行 *GetRandomNumber* 指令后，在内部更新 *RND_NUMBER* 寄存器的内容

表 25. RND_NUMBER 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|------------|---------|-----|
| b15-b0 | RND_NUMBER | 16 位随机数 | N/A |

提示

有关 *RND_NUMBER* 寄存器，请参见表 5. 系统寄存器列表。

5.1.2

密码管理

ST25TVxxxC 提供对用户和系统配置存储器的保护。通过基于密码的安全会话来控制对数据组的访问。在成功（对应失败）提供密码后，安全会话打开（对应关闭）并允许（对应拒绝）访问受保护数据组。

表 26. 安全会话类型

| 安全会话 | 通过显示密码打开 | 会话打开时授予的权限 |
|--------|----------|---------------------------------|
| CONFIG | PWD_CFG | 访问配置寄存器 更新 PWD_CFG |
| AREA1 | PWD_A1 | 访问用户存储器中的 AREA1 的块 更新 PWD_A1 |
| AREA2 | PWD_A2 | 访问用户存储器中的 AREA2 的块 更新 PWD_A2 |
| UNTR | PWD_UNTR | 更新 PWD_UNTR |

每个 PWD_CFG 和 PWD_UNTR 密码的长度为 32 位。

在双区域模式中（END_A1 < END_MEM），每个 PWD_A1 和 PWD_A2 密码的长度为 32 位。

在单区域模式中（END_A1 = END_MEM），PWD_A1 密码的长度为 64 位，AREA2 安全会话不适用：在使用单区域模式时，密码标识符为 02h 的密码指令失效。

提示

除了本节中所述的安全会话机制以外，PWD_CFG 和 PWD_UNTR 密码分别用于 *Kill* 和 *ToggleUntraceable* 指令。

表 27. 密码寄存器列表

| 密码 | Password_id | Password_data 大小 |
|----------|-------------|--|
| PWD_CFG | 00h | 4 字节 |
| PWD_A1 | 01h | 如果 END_A1 < END_MEM，则为 4 字节 如果 END_A1 = END_MEM，则为 8 字节 |
| PWD_A2 | 02h | 如果 END_A1 < END_MEM，则为 4 字节 如果 END_A1 = END_MEM，则为无效请求 |
| PWD_UNTR | 03h | 4 字节 |

ST25TVxxxC 密码管理基于三条指令：

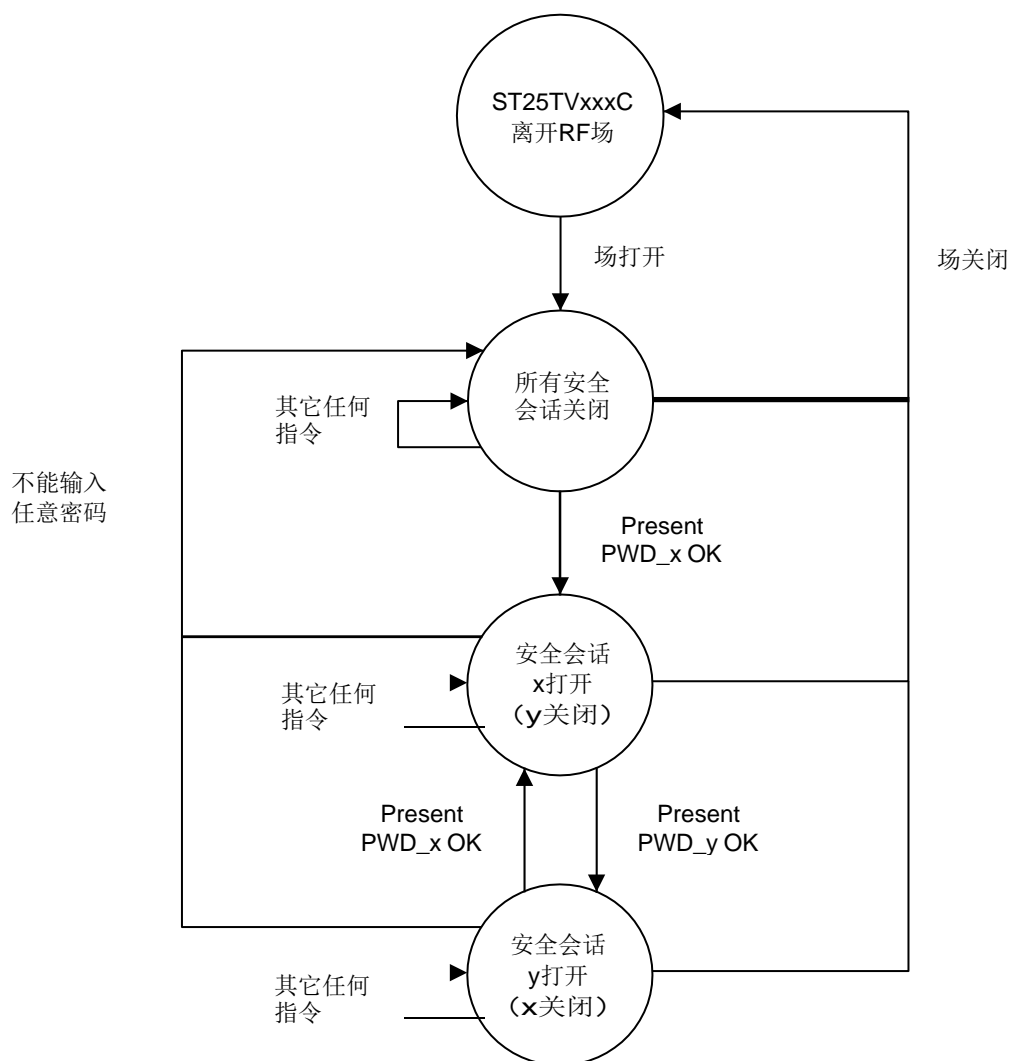
- WritePassword（参见第 6.4.18 节 WritePassword）
- PresentPassword（参见第 6.4.19 节 PresentPassword）
- GetRandomNumber（参见第 6.4.24 节 GetRandomNumber）

对于任何可用的 4x 密码，可采取三个动作：

- 打开安全会话：
 - 如果需要，使用 GetRandomNumber 指令
 - 使用具有相应的密码标识符与有效的加密密码值的 PresentPassword 指令（参见第 5.1.4 节 用户存储器保护）
- 更新密码：
 - 当相应密码的安全会话打开时，使用具有相同的密码标识符与新的加密密码值的 WritePassword 指令（参见第 5.1.4 节 用户存储器保护）。
- 关闭安全会话：
 - 要关闭与密码标识符对应的安全会话，用户可以选择以下选项之一：
 - 将标签移出 RF 场
 - 使用具有相同密码标识符与无效密码值的 PresentPassword 指令
 - 打开与不同的密码标识符对应的安全会话。打开一个新的安全会话会自动关闭先前打开的会话（即使打开操作失败）

图 6 描述打开/关闭安全会话的机制。

图 6. 安全会话管理



密码恢复

ST25TVxxxC 器件提供密码恢复功能，在 WritePassword 指令执行期间，当 RF 场失效时，这使用户能够对损坏的密码进行重新编程。

有关如何使用此功能的详细信息，请参见“AN5577 - ST25TV512C 和 ST25TV02KC 器件的密码管理”。请联系意法半导体销售办事处获取本文档。

密码尝试限制

借助失败密码尝试限制机制，ST25TVxxxC 器件提供密码防暴力攻击功能。

有关如何使用此功能的详细信息，请参见“AN5577 - ST25TV512C 和 ST25TV02KC 器件的密码管理”。请联系意法半导体销售办事处获取本文档。

5.1.3 密码加密

一种被称为封面编码的加密机制，用于传输以下指令帧的 `Password_data` 字段中的编码密码值：

- `PresentPassword` 请求（参见第 6.4.19 节 `PresentPassword`），
- `WritePassword` 请求（参见第 6.4.18 节 `WritePassword`）
- `Kill` 请求（参见第 6.4.20 节 `Kill`）
- `ToggleUntraceable` 请求（参见第 6.4.23 节 `ToggleUntraceable`）

该机制需要自上次启动 `ST25TVxxxC` 器件以来已发出 `GetRandomNumber` 指令调用，否则这些密码指令会失效。

此外，如果由于 `Password_data` 字段的值无效，而导致最后一次调用 `PresentPassword` / `Kill` / `ToggleUntraceable` 指令失败，则在对这三个指令中的任一指令尝试进行新调用时，必须发出 `GetRandomNumber` 指令调用，否则，无论 `Password_data` 字段的新值如何，其执行都将失败。

提示

如果 `PresentPassword` / `ToggleUntraceable` 指令的最后一次执行成功，则在发出新的 `PresentPassword` / `Kill` / `ToggleUntraceable` 请求之前，没必要发出新的 `GetRandomNumber` 指令调用。

假设已满足这些约束条件，则通过串联最后一次调用 `GetRandomNumber` 请求所返回的 `RND_NUMBER` 寄存器值来计算 `RND_NUMBER_4B` 和 `RND_NUMBER_8B` 值。

表 28. `RND_NUMBER_4B`

| b31-b16 | b15-b0 |
|-------------------------|-------------------------|
| <code>RND_NUMBER</code> | <code>RND_NUMBER</code> |

表 29. `RND_NUMBER_8B`

| b63-b48 | b47-b32 | b31-b16 | b15-b0 |
|-------------------------|-------------------------|-------------------------|-------------------------|
| <code>RND_NUMBER</code> | <code>RND_NUMBER</code> | <code>RND_NUMBER</code> | <code>RND_NUMBER</code> |

让 `PASSWORD_4B`（对应 `PASSWORD_8B`）成为 32 位（对应 64 位）密码的未加密值，将通过 `PresentPassword` / `WritePassword` / `Kill` / `ToggleUntraceable` 请求来传送该密码。

应按以下方式计算请求帧中的 `Password_data` 字段：

- 对于 32 位密码：
 - `Password_data = XOR(RND_NUMBER_4B, PASSWORD_4B)`
- 对于 64 位密码：
 - `Password_data = XOR(RND_NUMBER_8B, PASSWORD_8B)`

表 30. 64 位 `Password_data` 值计算示例

| | b63-b56 | b55-b48 | b47-b40 | b39-b32 | b31-b24 | b23-b16 | b15-b8 | b7-b0 |
|----------------------------|---------|---------|---------|---------|---------|---------|--------|-------|
| <code>RND_NUMBER</code> | - | - | - | - | - | - | 1Dh | E6h |
| <code>RND_NUMBER_8B</code> | 1Dh | E6h | 1Dh | E6h | 1Dh | E6h | 1Dh | E6h |
| <code>PASSWORD_8B</code> | FAh | D7h | 5Eh | 15h | CAh | A5h | D0h | D4h |
| <code>Password_data</code> | E7h | 31h | 43h | F3h | D7h | 43h | CDh | 32h |

提示

在 `ISO15693` 请求与响应帧中，以 `LSB` 到 `MSB` 字节顺序传输以多个字节编码的字段，如 `Password_data`

在处理 `PresentPassword` / `Kill` / `ToggleUntraceable` 请求时，`ST25TVxxxC` 器件解密 `Password_data` 字段，以获得 `PASSWORD_4B`（或 `PASSWORD_8B`）的未加密值，该值用于与由 `Password_id` 字段标识的密码寄存器进行比较。

在处理 `WritePassword` 请求时，`ST25TVxxxC` 器件解密 `Password_data` 字段，以获得 `PASSWORD_4B`（或 `PASSWORD_8B`）的未加密值，该值用于更新由 `Password_id` 字段标识的密码寄存器。

5.1.4 用户存储器保护

读和/或写访问保护可全局应用于区域块。借助 `RW_PROTECTION_A1` 和 `RW_PROTECTION_A2` 寄存器，可为 `AREA1` 和 `AREA2` 单独配置此类保护（参见表 11. `RW_PROTECTION_A1` 内容和表 15. `RW_PROTECTION_A2` 内容）。

工厂交付时，`AREA1` 和 `AREA2` 访问不受保护。更新 `RW_PROTECTION_Ax` 寄存器时，新保护模式在下一个 RF 会话的启动序列期间生效。

除了区域保护机制以外，借助 `LockBlock` 指令，可单独锁定由 `AREA1` 和 `AREA2` 组成的每个块的写访问。

块 00h 为区域保护机制的一个例外：

- 当块 00h 未锁定时，和 `AREA1` 的其他块一样，其写访问保护由 `RW_PROTECTION_A1` 寄存器的值决定
- 无论 `RW_PROTECTION_A1` 寄存器的值如何，始终允许对块 00h 的读访问

当 `LCK_A1` 寄存器设为 1b 时，`RW_PROTECTION_A1` 寄存器锁定。

当 `LCK_A2` 寄存器设为 1b 时，`RW_PROTECTION_A2` 寄存器锁定。

当 `LCK_A1` 和 `LCK_A2` 寄存器之一设为 1b 时，`END_A1` 寄存器锁定。

获取用户存储器块的安全状态

用户可以通过发出以下指令来读取块安全状态（BSS）：

- `GetMultipleBlockSecurityStatus`
- `Option_flag` 设为 1 的 `ReadSingleBlock`
- `Option_flag` 设为 1 的 `ReadMultipleBlocks`

对于每个块，根据 ISO 15693 标准中的规定，ST25TVxxxC 将以包含 `Lock_bit` 标志（表 31 中的 b0）的 BSS 字节响应。

表 31. 块安全状态

| b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|--------|----|----|----|----|----|----|---------------|
| 0: RFU | | | | | | | 0: 允许对当前块的写访问 |
| | | | | | | | 1: 拒绝对当前块的写访问 |

如果不允许对相应块的写访问，则该 `Lock_bit` 标志设为 1。当满足以下条件之一时，就会出现这种情况：

- 成功执行 `LockBlock` 指令后，对块的写访问永久锁定（`LCK_BLOCK` 寄存器的相应位设为 1b）
- 对父区域的写访问受到保护（RF 会话开始时，`RW_PROTECTION_Ax` = 01b 或 10b），安全会话关闭。
- 对父区域的写访问被禁止（RF 会话开始时，`RW_PROTECTION_Ax` = 11b）

5.1.5 系统配置存储器保护

使用 `ReadConfiguration` 和 `WriteConfiguration` 命令来访问列出的配置寄存器表 4. 配置寄存器列表。

按功能对配置寄存器进行分组。组由参数 `FID` 标识，该组中的寄存器由参数 `PID` 标识。

配置寄存器的写访问受保护或被禁止。

提示

只读配置寄存器的写访问被禁止

当 `CONFIG` 安全会话打开时，授予对配置寄存器的受保护写访问权限，其父组不会被永久锁定。

对配置寄存器的读访问受保护或始终允许。当 `CONFIG` 安全会话打开时，授予对配置寄存器的受保护读访问权限，其父组不会被永久锁定。

工厂交付时，配置组未锁定（`LCK_CONFIG` 寄存器的所有位设为 0b）。可通过将 `LCK_CONFIG` 寄存器的 `FID` 位设为 1b 来永久锁定通过 `FID`（00h、01h、02h、03h、04h、05h 或 08h）来识别的配置组：

- 如果对该组的配置寄存器的读访问受保护，即使 `CONFIG` 安全会话已打开，也无法再读取寄存器
- 如果对该组的配置寄存器的写访问受保护，即使 `CONFIG` 安全会话已打开，也无法再写入寄存器
- `CONFIG` 安全会话打开时，授予对 `LCK_CONFIG` 寄存器（`FID`=FFh，`PID`=00h）的写访问权限
- 即使在打开 `CONFIG` 安全会话后，用户也无法通过将 `LCK_CONFIG` 的 `FID` 位设回 0b 来解锁配置组（锁定是永久性的）
- 通过在请求中将 `LCK_CONFIG` 的相应位设为 1b，用户可通过单个 `WriteConfiguration` 指令锁定多个配置组

表 5. 系统寄存器列表中列出的系统寄存器包括密码、器件识别寄存器、锁定状态和指令状态。

除密码、AFI 和 DSFID 锁定状态、Kill 和 ToggleUntraceable 指令状态以外，可对系统寄存器进行读访问
第 5.10 节中详细介绍了器件识别寄存器：

- 可通过 LockAFI 和 LockDSFID 指令分别锁定对 AFI 和 DSFID 寄存器的写访问。锁定具有永久性：一旦锁定，对 AFI 与 DSFID 寄存器的写访问将被禁止。
- 其他器件识别寄存器（IC_REF、UID）是只读寄存器。

5.2 唯一输入码

5.2.1 唯一输入码寄存器

表 32. UTC_EN 访问

| RF 指令 | 访问类型 |
|--|---|
| ReadConfiguration @(FID=02h, PID=00h) | R: 始终可能 |
| WriteConfiguration @(FID=02h, PID=00h) | W: 如果 CONFIG 安全会话已打开且 LCK_UTC=0b W 生效时间：在下一个 RF 启动序列 |

表 33. UTC_EN 内容

| 位 | 名称 | 功能 | 出厂值 |
|-------|--------|----------------------------|----------|
| b0 | UTC_EN | 0: 唯一输入码被禁用 1: 唯一输入码已启用 | 0b |
| b7-b1 | RFU | - | 0000000b |

提示 有关 UTC_EN 寄存器，请参见表 4. 配置寄存器列表。

表 34. UTC 访问

| RF 指令 | 访问类型 |
|---------------------------------------|------------------------|
| ReadConfiguration @(FID=02h, PID=01h) | R: 始终可能 |
| - | W: 无写访问 ⁽¹⁾ |

1. 当 UTC_EN 设为 1b 时，在 RF 启动序列期间，从内部更新 UTC 寄存器的内容

表 35. UTC 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|-----|--------|-----|
| b23-b0 | UTC | 唯一输入码值 | 不适用 |

提示 有关 UTC 寄存器，请参见表 4. 配置寄存器列表。

5.2.2 唯一输入码描述

当 UTC_EN 寄存器设为 1b，在每个 RF 启动序列期间，在内部用新值更新 UTC 寄存器的内容。UTC 寄存器的内容以 ASCII 格式编码。

当 LCK_UTC 寄存器设为 1b 时，UTC_EN 寄存器锁定。

提示 当唯一输入码被使能时，RF 启动序列 t_{Boot_RF} 的持续时间（参见 8.2）为：

- 符合在 NFC Forum [DIGITAL] 规范中定义的 5ms 保护时间值
- 不符合在 ISO15693 规范中定义的 1ms 保护时间值

有关 UTC 寄存器内容的更多详细信息，请参见“AN5578 - ST25TV512C 和 ST25TV02KC 器件的唯一输入码”。请联系意法半导体销售办事处获取本文档。

5.3 篡改检测

篡改检测功能仅在 ST25TV02KC-T 器件上可用（参见第 10 节）。在 ST25TVxxx-A 器件上，通过 FID=03h 请求的 ReadConfiguration 与 WriteConfiguration 指令失败，错误代码为 10h。

5.3.1 篡改检测寄存器

表 36. TD_EVENT_UPDATE_EN 访问

| RF 指令 | 访问类型 |
|--|--|
| ReadConfiguration @(FID=03h, PID=00h) | R: 始终可能 |
| WriteConfiguration @(FID=03h, PID=00h) | W: 如果 CONFIG 安全会话打开且 LCK_TD=0b W 生效时间: 在下一个 RF 启动序列 |

表 37. TD_EVENT_UPDATE_EN 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|--------------------|----------------------------|----------|
| b0 | TD_EVENT_UPDATE_EN | 0: 禁用篡改事件记录 1: 启用篡改事件记录 | 0b |
| b7- b1 | RFU | - | 0000000b |

提示

有关 TD_EVENT_UPDATE_EN 寄存器，请参见表 4. 配置寄存器列表。

表 38. TD_SEAL_MSG 访问

| RF 指令 | 访问类型 |
|--|--|
| ReadConfiguration @(FID=03h, PID=01h) | R: 如果 CONFIG 安全会话打开且 LCK_TD=0b |
| WriteConfiguration @(FID=03h, PID=01h) | W: 如果 CONFIG 安全会话打开且 LCK_TD=0b W 生效时间: 立即 |

表 39. TD_SEAL_MSG 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|-------------|---------------------------------|-------|
| b15-b0 | TD_SEAL_MSG | TD_UNSEAL 事件首次发生前显示的 TD_EVENT 值 | 3030h |

注: 有关 TD_SEAL_MSG 寄存器，请参见表 4. 配置寄存器列表。

表 40. TD_UNSEAL_MSG 访问

| RF 指令 | 访问类型 |
|--|--|
| ReadConfiguration @(FID=03h, PID=02h) | R: 如果 CONFIG 安全会话打开且 LCK_TD=0b |
| WriteConfiguration @(FID=03h, PID=02h) | W: 如果 CONFIG 安全会话打开且 LCK_TD=0b W 生效时间: 立即 |

表 41. TD_UNSEAL_MSG 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|---------------|---------------------------------|-------|
| b15-b0 | TD_UNSEAL_MSG | TD_UNSEAL 事件首次发生后显示的 TD_EVENT 值 | 5555h |

提示

注：有关 TD_UNSEAL_MSG 寄存器，请参见表 4. 配置寄存器列表。

表 42. TD_RESEAL_MSG 访问

| RF 指令 | 访问类型 |
|--|--------------------------------|
| ReadConfiguration @(FID=03h, PID=03h) | R: 如果 CONFIG 安全会话打开且 LCK_TD=0b |
| WriteConfiguration @(FID=03h, PID=03h) | W: 如果 CONFIG 安全会话打开且 LCK_TD=0b |
| | W 生效时间: 立即 |

表 43. TD_RESEAL_MSG 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|---------------|-------------------------------|-------|
| b15-b0 | TD_RESEAL_MSG | TD_RESEAL 事件发生后显示的 TD_EVENT 值 | 5252h |

提示

注：有关 TD_RESEAL_MSG 寄存器，请参见表 4. 配置寄存器列表。

表 44. TD_SHORT_MSG 访问

| RF 指令 | 访问类型 |
|--|--------------------------------|
| ReadConfiguration @(FID=03h, PID=04h) | R: 如果 CONFIG 安全会话打开且 LCK_TD=0b |
| WriteConfiguration @(FID=03h, PID=04h) | W: 如果 CONFIG 安全会话打开且 LCK_TD=0b |
| | W 生效时间: 立即 |

表 45. TD_SHORT_MSG 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|--------------|------------------------------|-----|
| b7- b0 | TD_SHORT_MSG | 篡改回路在最后一个启动序列期间处于关闭状态时所显示的消息 | 63h |

提示

注：有关 TD_SHORT_MSG 寄存器，请参见表 4. 配置寄存器列表。

表 46. TD_OPEN_MSG 访问

| RF 指令 | 访问类型 |
|--|--------------------------------|
| ReadConfiguration @(FID=03h, PID=05h) | R: 如果 CONFIG 安全会话打开且 LCK_TD=0b |
| WriteConfiguration @(FID=03h, PID=05h) | W: 如果 CONFIG 安全会话打开且 LCK_TD=0b |
| | W 生效时间: 立即 |

表 47. TD_OPEN_MSG 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|-------------|------------------------------|-----|
| b7- b0 | TD_OPEN_MSG | 篡改回路在最后一个启动序列期间处于开启状态时所显示的消息 | 6Fh |

提示

有关 TD_OPEN_MSG 寄存器，请参见表 4. 配置寄存器列表。

表 48. TD_STATUS 访问

| RF 指令 | 访问类型 |
|---------------------------------------|---------|
| ReadConfiguration @(FID=03h, PID=06h) | R: 始终可能 |
| - | W: 无写访问 |

表 49. TD_STATUS 内容

| 位 | 名称 | 功能 | 出厂值 |
|---------|----------|--|-----|
| b15-b0 | TD_EVENT | TD_SEAL_MSG、TD_UNSEAL_MSG 或 TD_RESEAL_MSG 取决于篡改事件检测结果 | 不适用 |
| b23-b16 | TD_LOOP | TD_SHORT_MSG 或 TD_OPEN_MSG 取决于最后一个启动序列期间的篡改回路状态 | 不适用 |

提示

有关 TD_STATUS 寄存器，请参见表 5. 系统寄存器列表。

5.3.2

篡改检测说明

篡改检测功能可用于检查 ST25TV02KC-T 的 TD0 和 TD1 引脚之间的短路状态，并监控篡改事件。

有关打开与闭合的篡改回路的推荐阻抗值 R_{open} 和 R_{closed} ，请参见第 8.2 节 RF 电气参数。

在响应 FID=03h 且 PID=06h 的 ReadConfiguration 请求时，读取短路状态 TD_LOOP 与事件状态 TD_EVENT。

客户负责检查 TD_LOOP 和 TD_EVENT 的值，并相应地采取行动。

TD_LOOP

器件每次上电时，ST25TV02KC-T 捕获短路状态 TD_LOOP。TD_LOOP 的值等于以下值：

- TD_SHORT_MSG，TD0 和 TD1 在捕获时间连接时
- TD_OPEN_MSG，TD0 和 TD1 在捕获时间未连接时

此信息在掉电期间将丢失（TD_LOOP 未永久存储）。

TD_EVENT

TD_EVENT 状态用于监测 TD_UNSEAL 与 TD_RESEAL 事件的首次发生，这些事件的定义如下：

- TD_UNSEAL: TD_EVENT_UPDATE_EN 寄存器设为 1b，TD0 和 TD1 在捕获时间未连接
- TD_RESEAL: TD_EVENT_UPDATE_EN 寄存器设为 1b，TD_UNSEAL 已发生，TD0 和 TD1 在捕获时间已连接

工厂交付时，TD_EVENT_UPDATE_EN 寄存器设为 0b，TD_EVENT 的值设为

TD_SEAL_MSG 的值。

当 TD_UNSEAL 事件发生时，在 RF 启动序列期间，TD_EVENT 的值设为 TD_UNSEAL_MSG 的值。

当 TD_RESEAL 事件发生时，在 RF 启动序列期间，TD_EVENT 的值设为 TD_RESEAL_MSG 的值。

在掉电期间将保存该信息（TD_EVENT 永久存储）。

当 LCK_TD 寄存器设为 1b 时，TD_EVENT_UPDATE_EN、TD_SEAL_MSG、TD_UNSEAL_MSG、TD_RESEAL_MSG、TD_SHORT_MSG 和 TD_OPEN_MSG 寄存器锁定。

提示

当 TD_UNSEAL 或 TD_RESEAL 事件发生时，RF 启动序列 t_{Boot_RF} 的持续时间为（参见第 8.2 节 RF 电气参数）：

- 符合在 NFC Forum [DIGITAL] 规范中定义的 5ms 保护时间值
- 不符合在 ISO15693 规范中定义的 1ms 保护时间值

提示

当 TD_EVENT_UPDATE_EN 和 UTC_EN 寄存器设为 0b 时，在 RF 启动序列期间不对 EEPROM 进行编程，其持续时间符合在 ISO15693 规范中定义的 1ms 保护时间值。

提示

在捕获窗口之外发生的篡改检测事件（例如当 IC 处于 POWER-OFF 状态时，或在启动序列之后的 RF 会话期间）不会被 ST25TV02KC-T 检测到。

5.4 增强型 NDEF

5.4.1 增强型 NDEF 寄存器

表 50. ANDEF_EN 访问

| RF 指令 | 访问类型 |
|--|---|
| ReadConfiguration @(FID=04h, PID=00h) | R: 始终可能 |
| WriteConfiguration @(FID=04h, PID=00h) | W: 如果 CONFIG 安全会话打开且 LCK_ANDEF=0b W 生效时间: 在下一个 RF 启动序列 |

表 51. ANDEF_EN 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|----------|-------------------------------------|----------|
| b0 | ANDEF_EN | 0: 已禁用 ANDEF 特性, 1: 已启用 ANDEF 特性 | 0b |
| b7- b1 | RFU | - | 0000000b |

提示 有关 ANDEF_EN 寄存器, 请参见表 4. 配置寄存器列表。

表 52. ANDEF_CFG 访问

| RF 指令 | 访问类型 |
|--|---|
| ReadConfiguration @(FID=04h, PID=01h) | R: 始终可能 |
| WriteConfiguration @(FID=04h, PID=01h) | W: 如果 CONFIG 安全会话打开且 LCK_ANDEF=0b W 生效时间: 在下一个 RF 启动序列 |

表 53. ANDEF_CFG 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|----------------------------|--|-----|
| b0 | ANDEF_UID_EN | 0: 禁用 ANDEF 特性中的 UID 字段 1: 启用 ANDEF 特性中的 UID 字段 | 0b |
| b1 | ANDEF_CUS_EN | 0: 禁用 ANDEF 特性中的自定义字段 1: 启用 ANDEF 特性中的自定义字段 | 0b |
| b2 | ANDEF_UTC_EN | 0: 禁用 ANDEF 特性中的唯一输入码字段 1: 启用 ANDEF 特性中的唯一输入码字段 | 0b |
| b3 | RFU | - | 0b |
| b4 | ANDEF_TD_EN ⁽¹⁾ | 0: 禁用 ANDEF 特性中的篡改检测字段 1: 启用 ANDEF 特性中的篡改检测字段 | 0b |
| b5 | ANDEF_SEP_EN | 0: 禁用 ANDEF 字段分隔符 1: 启用 ANDEF 字段分隔符 | 1b |
| b7- b6 | ANDEF_BYTE | ANDEF 功能开始运行的 ANDEF_BLOCK 块中的字节偏移 | 00b |
| b15-b8 | ANDEF_BLOCK | ANDEF 功能开始运行的块地址 | 00h |

1. 仅在 ST25TV02KC-T 器件上相关, 在 ST25TVxxxC-A 器件上被强制为 0b

提示 有关 ANDEF_CFG 寄存器, 请参见表 4. 配置寄存器列表。

表 54. ANDEF_SEP 访问

| RF 指令 | 访问类型 |
|--|---|
| ReadConfiguration @(FID=04h, PID=02h) | R: 如果 CONFIG 安全会话打开且 LCK_ANDEF=0b |
| WriteConfiguration @(FID=04h, PID=02h) | W: 如果 CONFIG 安全会话打开且 LCK_ANDEF=0b W 生效时间: 立即 |

表 55. ANDEF_SEP 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|-----------|------------------------------------|-----|
| b7- b0 | ANDEF_SEP | ANDEF_SEP_EN=1b 时用作 ANDEF 字段分隔符的字符 | 78h |

提示 有关 ANDEF_SEP 寄存器, 请参见表 4. 配置寄存器列表。

表 56. ANDEF_CUSTOM_LSB 访问

| RF 指令 | 访问类型 |
|--|---|
| ReadConfiguration @(FID=04h, PID=03h) | R: 如果 CONFIG 安全会话打开且 LCK_ANDEF=0b |
| WriteConfiguration @(FID=04h, PID=03h) | W: 如果 CONFIG 安全会话打开且 LCK_ANDEF=0b W 生效时间: 立即 |

表 57. ANDEF_CUSTOM_LSB 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|------------------|---------------------|-----------|
| b31-b0 | ANDEF_CUSTOM_LSB | ANDEF 自定义字段的第 4 个字符 | 2E2E2E2Eh |

提示 有关 ANDEF_CUSTOM_LSB 寄存器, 请参见表 4. 配置寄存器列表。

表 58. ANDEF_CUSTOM_MSB 访问

| RF 指令 | 访问类型 |
|--|---|
| ReadConfiguration @(FID=04h, PID=04h) | R: 如果 CONFIG 安全会话打开且 LCK_ANDEF=0b |
| WriteConfiguration @(FID=04h, PID=04h) | W: 如果 CONFIG 安全会话打开且 LCK_ANDEF=0b W 生效时间: 立即 |

表 59. ANDEF_CUSTOM_MSB 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|------------------|----------------------|-----------|
| b31-b0 | ANDEF_CUSTOM_MSB | ANDEF 自定义字段的最后 4 个字符 | 2E2E2E2Eh |

提示 有关 ANDEF_CUSTOM_MSB 寄存器, 请参见表 4. 配置寄存器列表。

表 60. ANDEF_UID 访问

| RF 指令 | 访问类型 |
|---------------------------------------|-------------------------------------|
| ReadSingleBlock ReadMultipleBlocks | R: 如果 ANDEF_EN=1b 且 ANDEF_UID_EN=1b |
| - | W: 无写访问 |

表 61. ANDEF_UID 内容

| 位 | 名称 | 功能 | 出厂值 |
|---------|-----------|-----------------------|-------------------------|
| b127-b0 | ANDEF_UID | ANDEF 特性的 UID 字段中显示的值 | 以“E0”开始的 ASCII 格式中的 UID |

提示 有关 ANDEF_UID 寄存器，请参见表 5. 系统寄存器列表。

5.4.2

增强型 NDEF 描述

“增强型 NDEF”特性（ANDEF）是一种内容自动关联 NDEF 消息服务，允许标签响应动态内容，无需最终用户显式更新 EEPROM。

在最后一个 RF 启动序列期间，当 ANDEF_EN 寄存器的值为 1b 时（对应于 0b）时，特性启用（对应于禁用）。当功能特性启用时，字节地址范围为 ANDEF_START 到 ANDEF_END 的用户存储器数据被虚拟存储器 ANDEF_MEM 中的内容替代，以响应 ReadSingleBlock 与 ReadMultipleBlocks 请求。

提示 当 ANDEF 特性启用时，ReadSingleBlock 和 ReadMultipleBlocks 请求的 BSS 响应值未被修改

提示 ANDEF 特性对 WriteSingleBlock 指令无影响。当特性启用时，在跨越[ANDEF_START:ANDEF_END]范围的块上发出 WriteSingleBlock 指令，来自指令的数据被直接写入用户存储器，而不会被易失性存储器的内容替代。

表 62. ST25TV02KC 上的 ANDEF 特性被禁用时的块数据读取

| 块地址 | 块数据 ⁽¹⁾ | | | | 备注 |
|--------------------|--------------------|-------|-------|-------|--------------------------|
| | Byte0 | Byte1 | Byte2 | Byte3 | |
| 00h | UM000 | UM001 | UM002 | UM003 | UM 的第一个块 ⁽²⁾ |
| ... | ... | ... | ... | ... | 没有从 ANDEF_MEM 存储器读取字节。 |
| 4Fh ⁽³⁾ | UM316 | UM317 | UM318 | UM319 | UM 的最后一个块 ⁽²⁾ |

1. 块数据响应 ReadSingleBlock 与 ReadMultipleBlocks 请求。
2. UM 代表用户存储器。
3. 本例中使用了 ST25TV02KC 存储器容量。请注意，最后一个块地址为 ST25TV512C 器件上的 0Fh。

表 63. ST25TV02KC 上的 ANDEF 特性被启用时的块数据读取

| 块地址 | 块数据 ⁽¹⁾ | | | | 备注 |
|--------------------|--------------------|-------|-------|-------|--|
| | Byte0 | Byte1 | Byte2 | Byte3 | |
| 00h | UM000 | UM001 | UM002 | UM003 | UM 的第一个块 ⁽²⁾ |
| ... | ... | ... | ... | ... | - |
| 2Dh | UM180 | UM181 | UM182 | UM183 | 示例，其中 ANDEF_START=185 且 ANDEF_END=190 (UM 的 6 个字节 ⁽²⁾ 被 AM 替代 ⁽³⁾) |
| 2Eh | UM184 | AM000 | AM001 | AM002 | |
| 2Fh | AM003 | AM004 | AM005 | UM191 | |
| 30h | UM192 | UM193 | UM194 | UM195 | |
| ... | ... | ... | ... | ... | - |
| 4Fh ⁽⁴⁾ | UM316 | UM317 | UM318 | UM319 | UM 的最后一个块 ⁽²⁾ |

1. 块数据响应 ReadSingleBlock 与 ReadMultipleBlocks 请求。
2. UM 代表用户存储器
3. AM 代表 ANDEF_MEM 存储器
4. 本例中使用了 ST25TV02KC 存储器容量。请注意，最后一个块地址为 ST25TV512C 器件上的 0Fh。

字节地址 ANDEF_START 与 ANDEF_END 取决于最后一个 RF 启动序列期间的 ANDEF_CFG 寄存器的值：

- $ANDEF_START = ANDEF_BLOCK * 4 + ANDEF_BYTE$
- $ANDEF_END = \min(END_MEM * 4 + 3, ANDEF_START + ANDEF_LEN - 1)$

工厂交付时，ANDEF_EN 寄存器设为 0b，ANDEF_CFG 寄存器设为 0020h。

当 LCK_ANDEF 寄存器设为 1b 时，ANDEF_EN、ANDEF_CFG、ANDEF_SEP、ANDEF_CUSTOM_LSB 和 ANDEF_CUSTOM_MSB 寄存器锁定。

5.5 消费者隐私保护

Kill 和 Untraceable 功能提供 GDPR 所需的消费者隐私功能。

5.5.1 增强型 NDEF messagePrivacy 寄存器示例

表 65. KILL_CMD 访问

| RF 指令 | 访问类型 |
|-------|---------------------------------|
| - | R: 无读访问 |
| Kill | W: 如果 Kill 指令以 Error_flag=0b 响应 |
| | W 生效时间: 立即 |

表 66. KILL_CMD 内容

| 位 | 名称 | 功能 | 出厂值 |
|----|----------|-------------------------------------|-----|
| b0 | KILL_CMD | 0: 未成功执行 Kill 指令 1: 成功执行 Kill 指令 | 0b |

提示 有关 KILL_CMD 寄存器，请参见表 5. 系统寄存器列表。

表 67. UNTR_CMD 访问

| RF 指令 | 访问类型 |
|-------------------|--|
| - | R: 无读访问 |
| ToggleUntraceable | W: 如果 ToggleUntraceable 指令以 Error_flag=0b 响应 |
| | W 生效时间: 立即 |

表 68. UNTR_CMD 内容

| 位 | 名称 | 功能 | 出厂值 |
|----|----------|--|-----|
| b0 | UNTR_CMD | 0: ToggleUntraceable 上次成功执行, Address_flag=0b 1: ToggleUntraceable 上次成功执行, Address_flag=1b | 0b |

提示 有关 UNTR_CMD 寄存器，请参见表 5. 系统寄存器列表。

表 69. PRIVACY 访问

| RF 指令 | 访问类型 |
|--|----------------------------------|
| ReadConfiguration @(FID=05h, PID=00h) | R: 始终可能 |
| WriteConfiguration @(FID=05h, PID=00h) | W: 如果 CONFIG 安全会话打开且 LCK_PRIV=0b |
| | W 生效时间: 在下一个 RF 启动序列 |

表 70. PRIVACY 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|----------|---|-------|
| b1- b0 | UNTR_DFT | 00: UNTR_CMD=1b 时, 器件以 UNTRACEABLE 状态启动 01: 器件始终以 UNTRACEABLE 状态启动 10: 当 UNTR_CMD=1b 或篡改回路关闭时, 器件以 UNTRACEABLE 状态启动 11: 当 UNTR_CMD=1b 或篡改回路开启时, 器件以 UNTRACEABLE 状态启动 | 00b |
| b2 | DIS_INV | 0: Inventory 指令以 UNTRACEABLE 状态响应 1: Inventory 指令在 UNTRACEABLE 状态静默 | 0b |
| b3 | DIS_KILL | 0: 启用 Kill 指令 1: 禁用 Kill 指令 | 0b |
| b7- b4 | RFU | - | 0000b |

提示 有关 PRIVACY 寄存器, 请参见表 4. 配置寄存器列表。

5.5.2 Kill 功能描述

当 ST25TVxxxC 处于 KILLED 状态时, 所有传入的 RF 请求被忽略。

成功执行 Kill 指令后 (参见第 6.4.20 节 Kill), ST25TVxxxC 进入 KILLED 状态, 该指令会将 KILL_CMD 寄存器设为 1b。一旦 ST25TVxxxC 进入 KILLED 状态, 它只能在 POWER-OFF 与 KILLED 状态之间切换 (参见第 6.2.8 节 ISO15693 状态)。

当 DIS_KILL 寄存器在最后一个启动序列期间分别具有值 0/1b 时, Kill 指令被使能/禁用。DIS_KILL 寄存器的更新在下一个 RF 启动序列生效。

当 Kill 指令被禁用时, Kill 请求被忽略, ST25TVxxxC 无法进入 KILLED 状态。

工厂交付时, KILL_CMD 和 DIS_KILL 寄存器设为 0b。

当 LCK_PRIV 寄存器设为 1b 时, DIS_KILL 寄存器锁定。

5.5.3 不可追踪功能描述

当 ST25TVxxxC 处于 UNTRACEABLE 状态时, 除以下请求以外的所有传入 RF 请求均会被忽略:

- GetRandomNumber 与 ToggleUntraceable 请求 (参见第 6.4.23 节 ToggleUntraceable 与第 6.4.24 节 GetRandomNumber)
- Inventory 与 ReadSingleBlock (仅限块 00h) 请求, 如果 DIS_INV 寄存器的值在最后一个 RF 启动序列期间为 0b

成功执行通过 Address_flag=1b 请求的 ToggleUntraceable 指令后, ST25TVxxxC 进入 UNTRACEABLE 状态, 该指令将 UNTR_CMD 寄存器设为 1b。

成功执行通过 Address_flag=0b 请求的 ToggleUntraceable 指令后, ST25TVxxxC 离开 UNTRACEABLE 状态, 该指令将 UNTR_CMD 寄存器设为 0b。

在 RF 启动序列后, 如果 KILL_CMD 寄存器设为 0b 且满足以下任一条件, ST25TVxxxC 会进入 UNTRACEABLE 状态:

- UNTR_CMD 寄存器的值为 1b
- UNTR_DFT 寄存器的值为 01b
- UNTR_DFT 寄存器的值为 10b 且篡改回路关闭 (仅限 ST25TV02KC-T 器件)
- UNTR_DFT 寄存器的值为 11b 且篡改回路开启 (仅限 ST25TV02KC-T 器件)

请参见第 6.2.8 节 ISO15693 状态以获得更多信息。

提示 在 KILL_CMD=UNTR_CMD=0b 且 UNTR_DFT=10b 的 RF 启动序列之后:

- 如果篡改回路关闭, 则 ST25TV02KC-T 进入 UNTRACEABLE 状态
- 如果篡改回路打开, 则 ST25TV02KC-T 进入 READY 状态

提示 在 KILL_CMD=UNTR_CMD=0b 且 UNTR_DFT=11b 的 RF 启动序列之后:

- 如果篡改回路打开, 则 ST25TV02KC-T 进入 UNTRACEABLE 状态
- 如果篡改回路关闭, 则 ST25TV02KC-T 进入 READY 状态

提示 在 ST25TVxxxC-A 器件上, UNTR_DFT 寄存器的值 10b 和 11b 被解读为值 00b。

声称客户不可追溯性的原因如下：

- 在 NFC Forum 应用中，块 00h 包含无法识别客户的 CC 文件
- 用户块 01h 到 END_MEM - 包含无法在 UNTRACEABLE 状态下访问的客户数据
- 而在 UNTRACEABLE 状态中，UID 值用于 Inventory 的请求与响应帧，ReadSingleBlock 指令锁定（参见第 7.1 节 不可追踪的 UID），并且不允许识别客户

此外，用户可配置 ST25TVxxxC，以通过将 DIS_INV 寄存器设为 1b，忽略 UNTRACEABLE 状态下的 Inventory 与 ReadSingleBlock 请求。

DIS_INV 与 UNTR_DFT 寄存器的更新在下一个 RF 启动序列生效。工厂交付时，DIS_INV 与 UNTR_DFT 寄存器设为 0。

当 LCK_PRIV 寄存器设为 1b 时，UNTR_DFT 与 DIS_INV 寄存器锁定。

5.6 TruST25™数字签名

ST25TVxxxC 器件支持 TruST25 数字签名功能，该功能允许用户基于唯一的数字签名验证器件的身份。

TruST25 解决方案包括意法半导体部署的安全工业化进程和工具，用于生成、存储和检查器件中的签名。

有关详细的使用方法，请参见“ST25TV512C 与 ST25TV02KC 器件的 AN5580 - TruST25™数字签名”。请联系意法半导体销售办事处获取本文档。

5.7 AFI 保护

5.7.1 AFI 保护寄存器

表 71. AFI_PROT 访问

| RF 指令 | 访问类型 |
|--|--|
| ReadConfiguration @(FID=08h, PID=00h) | R: 始终可能 |
| WriteConfiguration @(FID=08h, PID=00h) | W: 如果 CONFIG 安全会话打开且 LCK_AFIP=0b W 生效时间: 在下一个 RF 启动序列 |

表 72. AFI_PROT 内容

| 位 | 名称 | 功能 | 出厂值 |
|--------|----------|---|----------|
| b0 | AFI_PROT | 0: WriteAFI 与 LockAFI 指令不依赖于 AREA1 安全会话 1: 当 AREA1 安全会话关闭时，WriteAFI 与 LockAFI 指令失效 | 0b |
| b7- b1 | RFU | - | 0000000b |

提示 有关 AFI_PROT 寄存器，请参见表 4. 配置寄存器列表。

5.7.2 AFI 保护说明

此功能允许通过 AREA1 安全会话来保护 WriteAFI 与 LockAFI 指令，并通过 AFI_PROT 寄存器来配置。

工厂交付时，AFI_PROT 寄存器设为 0b。AFI_PROT 寄存器设为 0b 时：

- 如果 LCK_AFI 寄存器设为 0b，则 WriteAFI 指令成功，否则失败
- 如果 LCK_AFI 寄存器设为 0b，则 LockAFI 指令成功，否则失败

AFI_PROT 寄存器设为 1b 时：

- 如果 AREA1 安全会话打开且 LCK_AFI 寄存器设为 0b，则 WriteAFI 指令成功，否则失败
- 如果 AREA1 安全会话打开且 LCK_AFI 寄存器设为 0b，则 LockAFI 指令成功，否则失败

当 LCK_AFIP 寄存器设为 1b 时，AFI_PROT 寄存器锁定。

5.8 Inventory Initiated

ST25TVxxxC 提供使用 **Initiate_flag** 易失性寄存器来改进移动标签的防冲突序列的特殊功能。该寄存器受 **Initiate** 指令控制（参见第 6.4.21 节 **Initiate**），它使 ST25TVxxxC 能够响应 **InventoryInitiated** 请求（参见第 6.4.22 节 **InventoryInitiated**）。

对于多个标签处于读卡器的 RF 场的应用，使用标准 **Inventory** 指令可能会错过标签。原因在于，防冲突序列执行全局树搜索，并在树的每个节点与叶子上调用指令。在最坏的情况下，**WC** 标签在被列为搜索叶之前需要等待长时间的延迟。标签在搜索期间进入读卡器的 RF 场可能使此类延迟进一步增加，**WC** 标签在被列入清单之前可能已离开场。

可通过在防冲突序列中将标准 **Inventory** 指令替换为自定义 **InventoryInitiated** 指令来改进此用例。当多个标签穿过读卡器的 RF 场时，防冲突序列被 **Initiate** 指令启动，此命令可发起范围内的一组标签。在发出 **Initiate** 指令后，**InventoryInitiated** 请求会被进入 RF 场的标签忽略，这些请求仅由一组启动的标签处理，从而限制了完成防冲突序列所需的时间。当启动的标签被列入清单时，该标签处于 **QUIET** 状态，以忽略其他 **InventoryInitiated** 请求。

一旦防冲突序列完成，读卡器将启动新的序列，该序列仅在前一序列中已进入 RF 场的标签上运行，以此类推。

5.9 设备识别寄存器

本节中描述的寄存器位于系统配置内存中。更多详细信息，请参见第 4.3 节。

表 73. LCK_DSFD 访问

| RF 指令 | 访问类型 |
|-----------|---------------------------------|
| - | R: 无读访问 |
| LockDSFID | W: 如果 LCK_DSFD=0b W 生效时间: 立即 |

表 74. LCK_DSFD 内容

| 位 | 名称 | 功能 | 出厂值 |
|----|----------|---|-----|
| b0 | LCK_DSFD | 0: 未成功执行 LockDSFID 指令 1: 成功执行 LockDSFID 指令 | 0b |

提示 有关 **LCK_DSFD** 寄存器，请参见表 5. 系统寄存器列表。

表 75. LCK_AFI 访问

| RF 指令 | 访问类型 |
|---------|---|
| - | R: 无读访问 |
| LockAFI | W: 如果 LCK_AFI=0b 且 (AFI_PROT=0b 或 AREA1 安全会话打开) W 生效时间: 立即 |

表 76. LCK_AFI 内容

| 位 | 名称 | 功能 | 出厂值 |
|----|---------|---|-----|
| b0 | LCK_AFI | 0: 未成功执行 LockAFI 指令 1: 成功执行 LockAFI 指令 | 0b |

提示 有关 **LCK_AFI** 寄存器，请参见表 5. 系统寄存器列表。

表 77. DSFID 访问

| RF 指令 | 访问类型 |
|-----------------------|--------------------|
| Inventory | R: 始终可能 |
| GetSystemInfo | R: 始终可能 |
| ExtendedGetSystemInfo | R: 始终可能 |
| Initiate | R: 始终可能 |
| InventoryInitiated | R: 始终可能 |
| WriteDSFID | W: 如果 LCK_DSFID=0b |
| | W 生效时间: 立即 |

表 78. DSFID 内容

| 位 | 名称 | 功能 | 出厂值 |
|-------|-------|-------------------------|-----|
| b7-b0 | DSFID | ISO/IEC 15693 数据存储格式标识符 | 00h |

提示 有关 DSFID 寄存器, 请参见表 5. 系统寄存器列表。

表 79. AFI 访问

| RF 指令 | 访问类型 |
|-----------------------|---|
| GetSystemInfo | R: 始终可能 |
| ExtendedGetSystemInfo | R: 始终可能 |
| WriteAFI | W: 如果 LCK_AFI=0b 且 (AFI_PROT=0b 或 AREA1 安全会话打开) |
| | W 生效时间: 立即 |

表 80. AFI 内容

| 位 | 名称 | 功能 | 出厂值 |
|-------|-----|-----------------------|-----|
| b7-b0 | AFI | ISO/IEC 15693 应用系列标识符 | 00h |

提示 有关 AFI 寄存器, 请参见表 5. 系统寄存器列表。

表 81. IC_REF 访问

| RF 指令 | 访问类型 |
|-----------------------|---------|
| GetSystemInfo | R: 始终可能 |
| ExtendedGetSystemInfo | R: 始终可能 |
| - | W: 无法访问 |

表 82. IC_REF 内容

| 位 | 名称 | 功能 | 出厂值 |
|-------|--------|---------------------|-----|
| b7-b0 | IC_REF | ISO/IEC 15693 IC 参考 | 08h |

提示 有关 IC_REF 寄存器, 请参见表 5. 系统寄存器列表。

表 83. REV 访问

| RF 指令 | 访问类型 |
|---------------------------------------|---------|
| ReadConfiguration @(FID=FEh, PID=00h) | R: 始终可能 |
| - | W: 无访问 |

表 84. REV 内容

| 位 | 名称 | 功能 | 出厂值 |
|-------|----|--------|-----|
| b7-b0 | 版本 | IC 版本号 | 00h |

提示 有关 REV 寄存器，请参见表 4. 配置寄存器列表。

表 85. UID 访问

| RF 指令 | 访问类型 |
|---------------------------------------|---------|
| Inventory | R: 始终可能 |
| GetSystemInfo | R: 始终可能 |
| ExtendedGetSystemInfo | R: 始终可能 |
| Initiate | R: 始终可能 |
| InventoryInitiated | R: 始终可能 |
| ReadConfiguration @(FID=FEh, PID=01h) | R: 始终可能 |
| - | W: 无访问 |

表 86. UID 内容

| 位 | 名称 | 功能 | 出厂值 |
|---------|-----|------------------------|-----------|
| b7-b0 | UID | ISO/IEC 15693 UID 字节 0 | IC 制造商序列号 |
| b15-b8 | | ISO/IEC 15693 UID 字节 1 | |
| b23-b16 | | ISO/IEC 15693 UID 字节 2 | |
| b31-b24 | | ISO/IEC 15693 UID 字节 3 | |
| b39-b32 | | ISO/IEC 15693 UID 字节 4 | |
| b47-b40 | | ISO/IEC 15693 UID 字节 5 | 08h |
| b55-b48 | | ISO/IEC 15693 UID 字节 6 | 02h |
| b63-b56 | | ISO/IEC 15693 UID 字节 7 | E0h |

提示 有关 UID 寄存器，请参见表 5. 系统寄存器列表。

6 RF 操作

器件遵循 ISO/IEC 15693 和 NFC Forum Type 5 标签射频功率和信号接口规范以及防冲突和传输协议。

器件通过 13.56 MHz 的电磁波载波进行通信，并通过接收信号幅度调制（ASK：幅移键控）来解调传入的数据。通过 1.6 Kbit/s（使用 1/256 脉冲编码模式）或 26 Kbit/s（使用 1/4 脉冲编码模式）的数据率对接收到的 ASK 波实施 10%或 100%的调制。

ST25TVxxxC 使用曼彻斯特编码（采用一个或两个 423 kHz 和 484 kHz 的副载波频率）并通过负载变化产生输出数据。ST25TVxxxC 通过以 6.6 Kbit/s（低数据率模式）和 26 Kbit/s（高数据率模式）的数据率传输数据。

6.1 RF 通信

6.1.1 访问 ISO/IEC 15693 设备

“读卡器”和 ST25TVxxxC 之间的对话按以下方式进行:

- 通过读卡器的 RF 工作场激活 ST25TVxxxC
- 通过读卡器传送指令 (ST25TVxxxC 检测载波幅度调制)
- ST25TVxxxC 使用负载调制传输响应。

这些操作使用如下所述的功率传输和通信信号接口。这种技术被称为 **RTE**（读卡器先发言）。

工作场

ST25TVxxxC 在表 173. RF 特性中所定义的最大和最小电磁场 H 值之间连续操作。读卡器必须在这些限制内产生工作场。

功率传输

通过 ST25TVxxxC 和读卡器中的耦合天线以 13.56 MHz 的无线电频率向 ST25TVxxxC 传输功率。在 ST25TVxxxC 天线上将读卡器的工作场转换为 AC 电压, 该电压经过整流、滤波和内部调节。在通信期间, 通过 ASK 解调器对该接收信号的幅度调制 (ASK) 进行解调

频率

ISO 15693 标准将工作场的载波频率 (f_c) 定义为 $13.56 \text{ MHz} \pm 7 \text{ kHz}$ 。

提示 在本文档中, f_C 符号用作 f_{CC} ($f_C=13.56\text{ MHz}$) 的标称值。

6.2 射频协议

6.2.1 协议说明

传输协议（或简称“协议”）定义了用于在 VCD（疏耦合设备）和 VICC（疏耦合集成电路卡）之间双向交换指令和数据的机制。它基于“VCD 先发言”的概念。器件用作 VICC。

这意味着，除非收到并对 VCD 发出的指令进行适当解码，否则 ST25TVxxxC 不会开始传输。该协议基于指令交换，并由 VCD 与 ST25TVxxxC 之间的请求/响应事务组成：

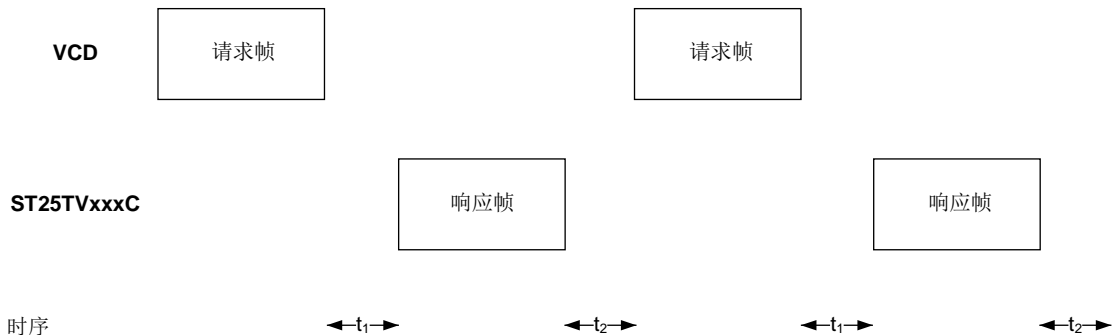
- 请求从 VCD 发送至 ST25TVxxxC
- 该请求的响应从 ST25TVxxxC 发送至 VCD。

每个请求和每个响应均包含在一个帧中。通过帧开头（SOF）和帧末尾（EOF）来分隔帧。

该协议面向比特。帧中发送的位数是八（8）的倍数，即整数个字节。

单字节字段先发送最低有效位（LSBit）。多字节字段先发送最低有效字节（LSByte），并且每个字节先发送最低有效位（LSBit）。

图 8. ISO15693 协议时序



6.2.2

请求格式

请求帧存在于：

- SOF
- 请求标志
- 指令代码
- 请求参数与数据
- CRC
- EOF。

表 87. 一般请求格式

| SOF | Request_flags | 操作码 | 参数 | 数据 | CRC_B | EOF |
|-----|---------------|-----|----|----|-------|-----|
| - | 8 位 | 8 位 | 可选 | 可选 | 16 位 | - |

6.2.3

请求标志

在请求帧中，Request_flags 字段指定了将由 ST25TVxxxC 执行的操作，以及是否存在相应的字段。

Request_flags 字段包含从 0 索引到 7 的 8 个位。

提示

位索引从 0 开始，并符合此规范中使用的惯例，需要注意的是，在 ISO/IEC 15693 规范中，这些位的索引从 1 开始。

Request_flags 的位 2 (Inventory_flag) 定义了 4 个 MSB (位 4 到 7) 的内容。

当 Inventory_flag 值为 0 时，位 4 到 7 定义了 ST25TVxxxC 选择标准。

当 Inventory_flag 值为 1 时，位 4 到 7 定义了 ST25TVxxxC Inventory 参数。

表 88. Request_flags LSB 的定义

| 位 | 标志 | 说明 |
|---|--------------------------------|--|
| 0 | Subcarrier_flag ⁽¹⁾ | 0: VICC 使用一个副载波 1: VICC 使用两个副载波 |
| 1 | Datarate_flag ⁽¹⁾ | 0: VICC 使用低数据率 1: VICC 使用高数据率 |
| 2 | Inventory_flag | 0: 位 4 至 7 如 所述 表 89 1: 位 4 至 7 如 所述 表 90 |
| 3 | Protocol_extension_flag | 0: 不存在协议格式扩展 1: 不支持 (RFU) |

1. Subcarrier_flag 和 Datarate_flag 是指 VICC-to-VCD 通信。

表 89. Inventory_flag 值为 0 时的 Request_flags MSB 定义

| 位 | 标志 | 说明 |
|---|-----------------------------|---|
| 4 | Select_flag ⁽¹⁾ | 0: 根据 Address_flag 的值来处理该指令 1: UID 字段不存在。该指令仅由 SELECTED 状态中的 VICC 来处理 ⁽²⁾ |
| 5 | Address_flag ⁽¹⁾ | 0: UID 字段不存在。指令由任意 VICC 处理 1: UID 字段存在。指令仅由 VICC 处理, 其 UID 与字段值匹配 |
| 6 | Option_flag | 0: Option 未激活 1: Option 激活 |
| 7 | RFU_flag | 0: 除非另有规定 1: 不支持 (RFU) |

1. Select_flag=1 与 Address_flag=1 为无效用例, 此类设置的请求会被 ST25TVxxx C 器件忽略。

2. SELECTED 状态在第 6.2.8 节中定义

表 90. Inventory_flag 值为 1 时的 Request_flags MSB 定义

| 位 | 标志 | 说明 |
|---|---------------|-------------------------------|
| 4 | AFI_flag | 0: AFI 字段不存在 1: AFI 字段存在 |
| 5 | Nb_slots_flag | 0: 16 时隙模式 1: 1 时隙模式 |
| 6 | Option_flag | 0: Option 未激活 1: Option 激活 |
| 7 | RFU_flag | 0: 除非另有规定 1: 不支持 (RFU) |

6.2.4

响应格式

响应帧存在于:

- SOF
- 响应标志
- 响应数据
- CRC
- EOF

表 91. 一般响应格式

| SOF | Response_flags | Response_data | CRC_B | EOF |
|-----|----------------|---------------|-------|-----|
| - | 8 位 | 可选 | 16 位 | - |

6.2.5

响应标志

在响应帧中，Response_flags 字段表示 ST25TVxxxC 如何执行操作，以及是否存在相应的字段。

Response_flags 字段包含从 0 索引到 7 的 8 个位。

提示

位索引从 0 开始，并符合此规范中使用的惯例，需要注意的是，在 ISO/IEC 15693 规范中，这些位的索引从 1 开始。

表 92. Response_flags 的定义

| 位 | 标志 | 说明 |
|---|------------|--------------------------------|
| 0 | Error_flag | 0: 无错误 1: 检测到错误。数据字段中存在错误代码 |
| 1 | RFU | 0: 除非另有规定 1: 不支持 (RFU) |
| 2 | | |
| 3 | | |
| 4 | | |
| 5 | | |
| 6 | | |
| 7 | | |

6.2.6

响应和错误码

如果 ST25TVxxxC 将响应中的 Error_flag 字段设为 1，则显示错误代码字段，并提供发生错误的相关信息。

如果在处理指令时发生错误，则 ST25TxxxC 保持静默，而不是在以下情况下响应将 Error_flag 设为 1 的帧：

- Inventory_flag 设为 1
- Inventory_flag、Select_flag 和 Address_flag 设为 0

表 93 中未指定的操作代码字段是为将来使用所预留的错误代码。

表 93. Error_flag 等于 1 时的一般响应格式

| SOF | Response_flags | Error_code | CRC_B | EOF |
|-----|----------------|------------|-------|-----|
| - | 01h | 8 位 | 16 位 | - |

表 94. 响应错误代码定义

| 错误代码 | 说明 |
|------|--------------------|
| 01h | 无效 IC Mfg 代码值 |
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 0Fh | 出错，未提供信息 |
| 10h | 请求数据不可用 |
| 11h | 请求数据已锁定，因此无法再次锁定 |
| 12h | 请求数据已锁定，无法更改其内容 |

| 错误代码 | 说明 |
|------|--------------|
| 13h | 对请求数据进行编程时失败 |
| 14h | 锁定请求数据失败 |
| 15h | 请求数据受读保护 |

6.2.7

模式

“模式”一词是指在指令中使用的机制，用于指定一组 VICC 器件，这组器件必须能处理 `Inventory_flag` 设为 0 的请求。根据在第 6.2.3 节 请求标志中定义的 `Address_flag` 与 `Select_flag` 值，定义了三种模式。

寻址模式（Addressed mode）

当 `Address_flag` 设为 1 时（寻址模式），请求包含寻址的 VICC 的 UID（唯一 ID）。

收到请求的任何 ST25TVxxxC（`Address_flag` 设为 1）将收到的 UID 与其自身进行比较。如果匹配，则器件处理请求（如果可能），并按照指令描述向 VCD 返回响应。否则，器件保持静默。

选择模式（Select mode）

当 `Select_flag` 设为 1 时（选择模式），请求帧不包含 UID 字段。仅处于 **SELECTED** 状态的 VICC 接收并处理请求（`Select_flag` 设为 1），并按照指令描述向 VCD 返回响应。

SELECTED 状态在第 6.2.8 节中定义。系统设计确保每次只有一个 ST25TVxxxC 可处于 **SELECTED** 状态。

非寻址模式（广播请求）

当 `Address_flag` 和 `Select_flag` 设为 0 时（非寻址模式），请求帧不包含 UID 字段。

在非寻址模式中，多个 VICC 可响应请求，与寻址与选择模式不同，最多一个 VICC 将响应请求。

6.2.8

ISO15693 状态

- POWER-OFF
- READY
- QUIET
- SELECTED

图 9. ISO15693 状态转换图说明了这些状态之间的转换。

POWER-OFF 状态

当无法从 VCD 获得足够的能量时，ST25TVxxxC 处于 RF POWER-OFF 状态。

READY 状态

当从 VCD 获得足够的能量时，ST25TVxxxC 以 **READY** 状态启动。

当处于 **READY** 状态时，ST25TVxxxC 处理寻址模式、非寻址模式或 `Inventory_flag` 设为 1 的请求。选择模式中的请求被忽略。

QUIET 状态

当处于 **QUIET** 状态时，ST25TVxxxC 处理寻址模式中的任意请求。选择或非寻址模式中的请求被忽略（除了非寻址模式中的 `ResetToReady` 指令）。`Inventory_flag` 设为 1 的请求被忽略。

SELECTED 状态

在 **SELECTED** 状态中，ST25TVxxxC 处理任意寻址模式中的请求：

- 选择模式中的请求
- 寻址模式中的请求
- 非寻址模式中的请求
- `Inventory_flag` 设为 1 的请求

表 95. 取决于寻址模式的 Request_flags 值

| Request_flags | 非寻址 | Select | 寻址 ⁽¹⁾ | Inventory ⁽¹⁾ |
|----------------|-----|--------|-------------------|--------------------------|
| Inventory_flag | 0 | 0 | 0 | 1 |
| Select_flag | 0 | 1 | 0 | - |
| Address_flag | 0 | 0 | 1 | - |

1. 假设 UID 与 Inventory 参数值与 ST25TVxxxC 寄存器值匹配

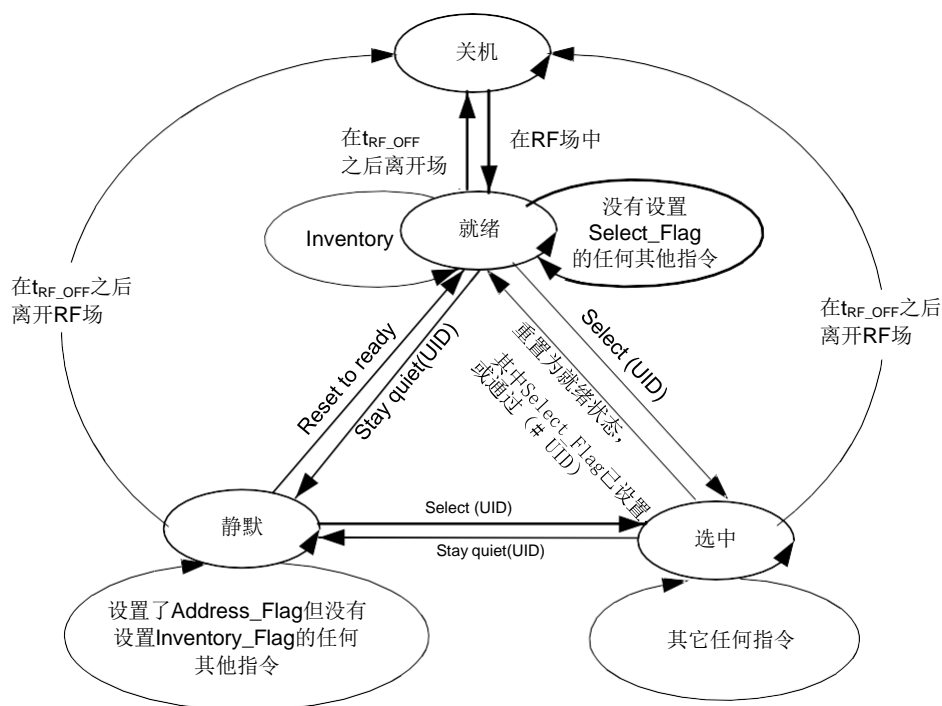
表 96. 取决于状态与寻址模式的器件响应

| ISO15693 状态 | 非寻址 | Select | 寻址 ⁽¹⁾ | Inventory ⁽¹⁾ |
|-------------|------|--------|-------------------|--------------------------|
| READY | X | - | X | X |
| SELECTED | X | X | X | X |
| QUIET | -(2) | - | X | - |

1. 假设 UID 与 Inventory 参数值与 ST25TVxxxC 寄存器值匹配

2. 除非寻址 ResetToReady 请求以外，在 QUIET 状态下将忽略所有非寻址请求

图 9. ISO15693 状态转换图



如果标签离开场至少 t_{RF_OFF} ，ST25TVxxxC 将返回 POWER-OFF 状态。此状态转换方法的目的是确保在任何给定时间只有一个 ST25TVxxxC 处于 SELECTED 状态。

6.2.9 自定义状态

除了上一节中所述的 ISO15693 状态以外，ST25TVxxxC 还支持两种自定义状态：

- UNTRACEABLE
- KILLED

图 10 中指定了这些状态之间的转换。

UNTRACEABLE 状态

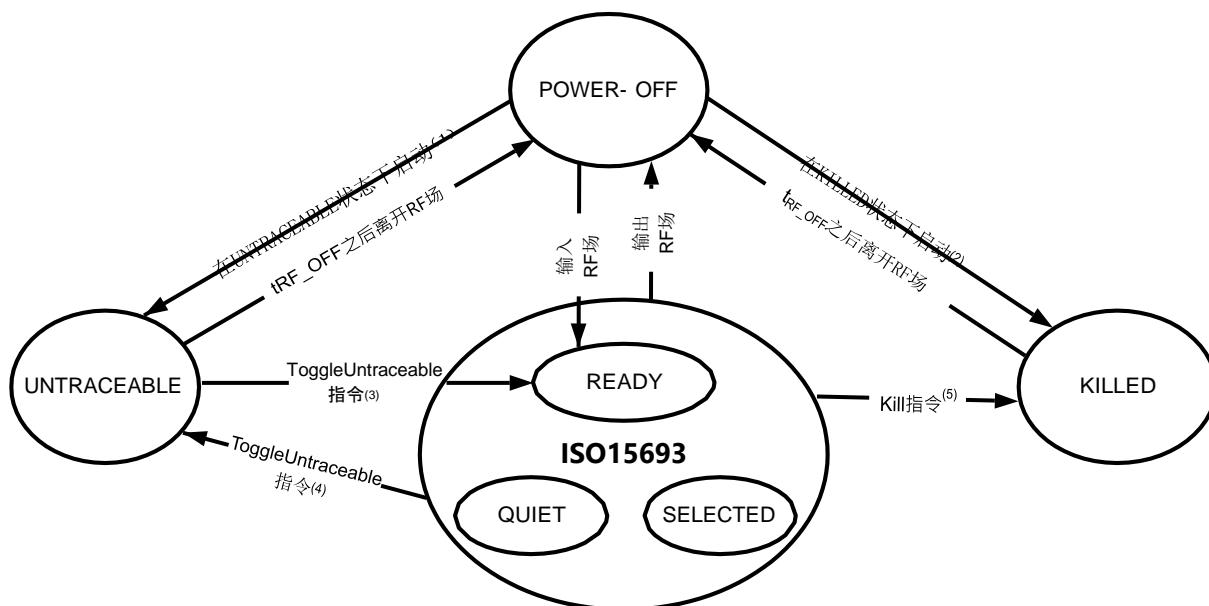
当处于 UNTRACEABLE 状态时，ST25TVxxxC 忽略除以下请求以外的所有传入请求：

- 非寻址模式下的 GetRandomNumber 请求
- 非寻址模式下的 ToggleUntraceable 请求
- Inventory 请求（如果 DIS_INV 寄存器的值在启动时为 0b）
- 寻址模式下的 ReadSingleBlock 请求（如果 DIS_INV 寄存器的值在启动时为 0b 且 Block_number 参数设为 00h）

KILLED 状态

当处于 KILLED 状态时，ST25TVxxxC 忽略所有传入请求。

图 10. ST25TVxxxC 状态转换图



- 在以下情况下，ST25TVxxxC 以 UNTRACEABLE 状态启动：
 - 在 RF 场中
 - KILL_CMD 寄存器的值为 0b
 - 满足以下任一条件：
 - UNTR_CMD 寄存器的值为 1b
 - UNTR_DFT 寄存器的值为 01b
 - UNTR_DFT 寄存器的值为 10b，篡改回路关闭
 - UNTR_DFT 寄存器的值为 11b，篡改回路开启
- 在以下情况下，ST25TVxxxC 以 KILLED 状态启动：
 - 在 RF 场中
 - KILL_CMD 寄存器的值为 1b
- 在成功执行以非寻址模式请求的 ToggleUntraceable 指令后（该操作将 UNTR_CMD 寄存器设为 0b），ST25TVxxxC 从 UNTRACEABLE 状态切换至 READY 状态
- 在成功执行以寻址模式请求的 ToggleUntraceable 指令后（该操作将 UNTR_CMD 寄存器设为 1b），ST25TVxxxC 成功从 READY/SELECTED/QUIET 状态切换至 UNTRACEABLE 状态
- 在成功执行以寻址模式请求的 Kill 指令后（该操作将 KILL_CMD 寄存器永久设为 1b），ST25TVxxxC 从 READY/SELECTED/QUIET 状态切换至 KILLED 状态

当 ST25TVxxxC 以 UNTRACEABLE 状态启动时，UID 寄存器的值被在第 7.1 节中指定的不可追踪的 UID 值屏蔽（除非在响应 ReadConfiguration 请求时（FID=FEh，PID=01h），始终返回 UID 寄存器的内容而不屏蔽），直至其恢复 POWER-OFF 状态，

当 ST25TVxxxC 以 READY 状态启动，并通过显式 ToggleUntraceable 命令进入 UNTRACEABLE 状态时，UID 寄存器的值被在第 7.1 节 不可追踪的 UID 中指定的不可追踪的 UID 值屏蔽，直至其离开 UNTRACEABLE 状态。

当 ST25TVxxxC 进入 UNTRACEABLE 状态时，在离开 UNTRACEABLE 状态之前：

- AFI 寄存器的值被 00h 屏蔽
- DSFID 寄存器的值被 00h 屏蔽

提示

当 UID 和/或 AFI 寄存器被屏蔽时，必须使用以下结果值：

- 在 Inventory_flag=1b 的 Mask_value 和 AFI 请求参数下
- 在 Inventory_flag=0b 且 Address_flag=1b 的 UID 请求参数下

6.3 时序定义

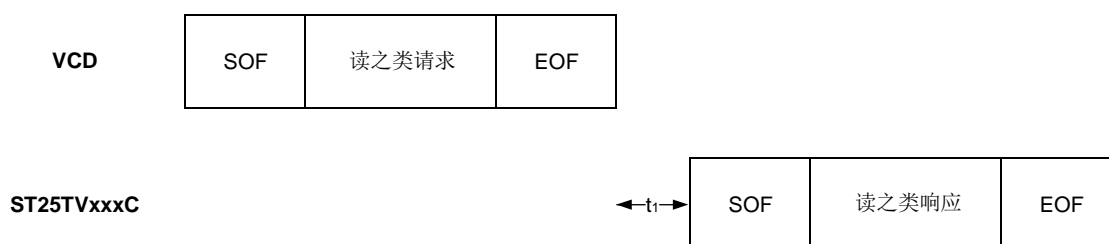
提示

特定时序的容差为 $\pm 32/f_C$

t₁：读之类指令的 VICC 响应延迟

对于读之类指令，例如不调用 EEPROM 编程的指令，从 VCD 收到请求后，在发送响应之前，VICC 的等待时间为 t₁，该时间从 EOF 的上升沿开始。值 t₁ 在表 97. 时序值中提供。

图 11. VCD 与 ST25TVxxxC 之间的读之类帧交换



t₂：VCD 新请求延迟

在 t₂ 时间后，VCD 可以发送 EOF，以便在 Inventory 请求后（Nb_slots_flag 设为 0）接收到一个或多个 VICC 响应时切换至下一个时隙。t₂ 时间从接收到来自 VICC 的 EOF 开始。

无论向 VICC 发送请求时使用了哪种调制指数，VCD 所发送的 EOF 均可以经过 10%或 100%调制。

t₂ 也是 VCD 向 VICC 发送新请求所需的等待时间，如图 8. ISO15693 协议时序中所述。

值 t₂ 在表 97. 时序值中提供。

t₃：收到 VICC 响应时的 VCD 新请求延迟

在 t₃ 时间后，VCD 可以发送 EOF，以便在 Inventory 请求后（Nb_slots_flag 设为 0），未从 VICC 接收到响应时切换至下一个时隙。

无论向 VICC 发送请求时使用了哪种调制指数，VCD 所发送的 EOF 均可以经过 10%或 100%调制。

从 VCD 所发送的请求 EOF 的上升沿开始：

- 若 EOF 经过 100%调制，则在发送新的 EOF 之前，VCD 将至少等待相当于 t_{3min} 的时间，以便进行 100%调制。
- 若 EOF 经过 10%调制，则在发送新的 EOF 之前，VCD 将至少等待相当于 t_{3min} 的时间，
- 以进行 10%调制。

表 97. 时序值

| | 最小 (min) 值 | | 标称 (nom) 值 | 最大 (max) 值 |
|-----------|----------------------------------|---|----------------------------|----------------------------------|
| | 100%调制 | 10%调制 | | |
| t_1 | $4320 / f_C = 318.6 \mu s$ | | $4352 / f_C = 320.9 \mu s$ | $4384 / f_C = 323.3 \mu s^{(1)}$ |
| t_2 | $4192 / f_C = 309.2 \mu s$ | | 无 | 无 |
| t_3 | $t_{1max}^{(2)} + t_{SOF}^{(3)}$ | $t_{1max}^{(2)} + t_{NRT}^{(4)} + t_{2min}$ | No t_{3nom} | No t_{3max} |
| t_{EOF} | 10 ms | 无 | t_{EOFnom} | 20 ms |

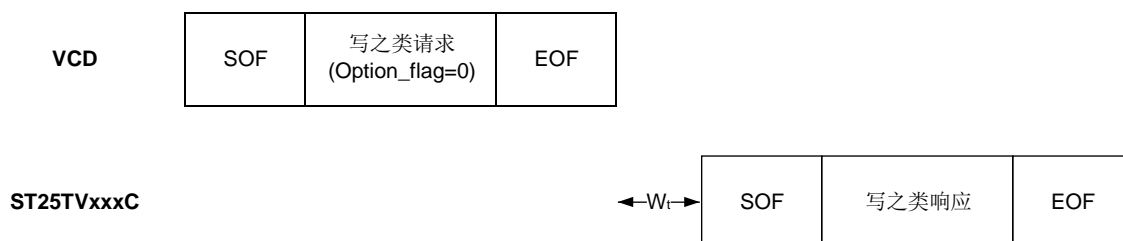
1. 在场上升之后的第一个毫秒期间，不会解析 VCD 的请求。
2. t_{1max} 不适用于写入类指令。由 W_t 和 t_{EOF} (见下文)
3. t_{SOF} 定义的写之类指令的特定时序约束是 VICC 将 SOF 发送至 VCD 所花费的时间。 t_{SOF} 取决于响应数据率：高数据率或低数据率。
4. t_{NRT} 为 VICC 的标称响应时间。 t_{NRT} 取决于响应数据率、副载波调制模式和预期响应帧的大小。

W_t : Option_flag=0 的写之类指令的 VICC 响应延迟

对于 option_flag=0 的写之类指令，例如涉及 EEPROM 编程的指令，从 VCD 收到请求后，在发送响应之前，VICC 的等待时间为 W_t ，该时间从 EOF 的上升沿开始。

W_t 时间等于 $t_{1nom} + 4096 / f_C (= 302 \mu s)$ 的倍数。

图 12. Option_flag=0 时的 VCD 与 ST25TVxxxC 之间的写之类帧交换



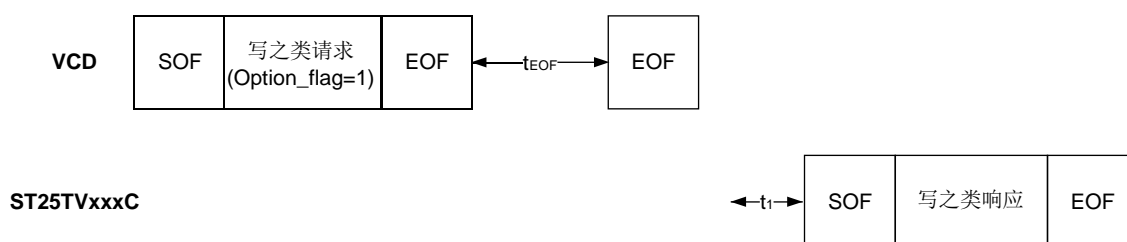
t_{EOF} : Option_flag=1 的写之类指令的 EOF 请求延迟

对于 Option_flag=1 的写之类指令，在发送隔离 EOF 请求前（将触发 VICC 的响应），VCD 的等待时间为 t_{EOF} ，该时间从请求帧中的 EOF 的上升沿开始。

在收到隔离 EOF 请求后，在发送响应之前，VICC 的等待时间为 t_1 ，该时间从隔离 EOF 请求的上升沿开始。

t_{EOF} 的容许值在表 97. 时序值中提供。

图 13. Option_flag=1 时 VCD 与 ST25TVxxxC 之间的写之类帧交换



6.4

RF 指令

ST25TVxxxC 支持以下 RF 指令集：

- Inventory, 用于执行防冲突序列。
- StayQuiet, 用于将 ST25TVxxxC 置于 QUIET 状态, 该状态只响应寻址模式下的指令。
- ReadSingleBlock, 用于读取块的 32 位及其锁定状态。
- WriteSingleBlock, 用于在授予写访问权限的情况下, 为 32 位块的更新写入和验证新内容。
- LockBlock, 用于永久禁止对所选块的写访问。
- ReadMultipleBlocks, 用于读取一系列块的内容及其锁定状态。
- Select, 用于将 ST25TVxxxC 置于 SELECTED 状态。在该命令之后, ST25TVxxxC 处理通过 Select_flag 设置请求的所有指令。
- ResetToReady, 用于将 ST25TVxxxC 置于 READY 状态。
- WriteAFI, 用于将 8 位值写入 AFI 寄存器中。
- LockAFI, 用于锁定 AFI 寄存器。
- WriteDSFID, 用于将 8 位值写入 DSFID 寄存器中。
- LockDSFID, 用于锁定 DSFID 寄存器。
- GetSystemInfo 与 ExtendedGetSystemInfo, 用于读取标准系统信息值。
- GetMultipleBlockSecurityStatus, 用于读取范围块的安全状态。
- ReadConfig, 用于读取配置寄存器。
- WriteConfig, 用于写入配置寄存器。
- Kill, 用于通过进入 KILLED 状态来永久禁用标签。
- WritePassword, 用于更改已打开的安全会话的密码。
- PresentPassword, 用于打开安全会话。
- GetRandomNumber, 用于生成 16 位编号。
- ToggleUntraceable, 用于进入或离开 UNTRACEABLE 状态。
- Initiate, 用于将 Initiate_flag 寄存器设为 1。
- InventoryInitiated, 用于在 Initiate_flag 设为 1 的 ST25TVxxxC 上执行防冲突序列。

表表 98 中提供了其代码。

表 98. 指令代码

| 操作码 | 指令 | 操作码 | 指令 |
|-----|--------------------|-----|--------------------------------|
| 01h | Inventory | 2Bh | GetSystemInfo |
| 02h | StayQuiet | 2Ch | GetMultipleBlockSecurityStatus |
| 20h | ReadSingleBlock | 3Bh | ExtendedGetSystemInfo |
| 21h | WriteSingleBlock | A0h | ReadConfig |
| 22h | LockBlock | A1h | WriteConfig |
| 23h | ReadMultipleBlocks | A6h | Kill |
| 25h | Select | B1h | WritePassword |
| 26h | ResetToReady | B3h | PresentPassword |
| 27h | WriteAFI | B4h | GetRandomNumber |
| 28h | LockAFI | BAh | ToggleUntraceable |
| 29h | WriteDSFID | D1h | InventoryInitiated |
| 2Ah | LockDSFID | D2h | Initiate |

6.4.1

Inventory

收到 Inventory 请求时, 如果参数与 UID 和 AFI 寄存器的值匹配, ST25TVxxxC 发送响应。

Inventory_flag 设为 1: Request_flags 的位 4 和 5 分别对 AFI_flag 和 Nb_slots_flag 进行编码。

Option_flag 设为 0: 没有支持的选项。

表 99. Inventory 请求格式

| SOF | Request_flags | 操作码 | AFI ⁽¹⁾ | Mask_length | Mask_value | CRC_B | EOF |
|-----|---------------|-----|--------------------|-------------|------------|-------|-----|
| - | 00xx01xxb | 01h | 8 位 | 8 位 | 0-64 位 | 16 位 | - |

1. 当 Request_flags=00x101xxb 时, AFI 字段存在

请求参数与数据包括:

- AFI_flag 设为 1 时的 AFI 参数
- 以位为单位的 Mask_length 长度, Nb_slots_flag = 0b 时 ≤ 60, Nb_slots_flag = 1b 时 ≤ 64
- Mask_value, 以字节为单位的大小为(Mask_length + 7)/8, 如果 Mask_length = 00h, 则不存在

表 100. Inventory 响应格式

| SOF | Response_flags | DSFID | UID | CRC_B | EOF |
|-----|----------------|-------|------|-------|-----|
| - | 00h | 8 位 | 64 位 | 16 位 | - |

当 Error_flag 设为 0 时, 响应数据包括:

- DSFID 寄存器值
- UID 寄存器值

ST25TVxxxC 在出错时不会产生任何应答。

当 VICC 响应 Inventory 请求时, 帧交换的时序为图 11. VCD 与 ST25TVxxxC 之间的读之类帧交换中所述的读之类指令的时序。

当 Nb_slots_flag 设为 0 时, 在表 99. Inventory 请求格式中的初始请求后 (包含第 6.3 节 时序定义中所述的以下时序), VCD 发出 15 EOF 请求:

- 如果 VICC 响应 EOF 请求, 则帧交换的时序为读之类指令的时序
- 如果 VCD 从一个或多个 VICC 收到响应, 则在发送下一个 EOF 请求前, 其等待时间为 t_2
- 如果 VCD 未从任何 VICC 收到响应, 则在发送下一个 EOF 请求前, 其等待时间为 t_3

6.4.2

StayQuiet

收到 StayQuiet 请求时:

- 如果没有出现错误, ST25TVxxxC 进入 QUIET 状态, 也不会返回响应。
- 即使发生错误, 也不会响应 StayQuiet 指令。

Select_flag 设为 0, Address_flag 设为 1: StayQuiet 请求必须在寻址模式下发出。

Option_flag 设为 0: 没有支持的选项。

表 101. StayQuiet 请求格式

| SOF | Request_flags | 操作码 | UID | CRC_B | EOF |
|-----|---------------|-----|------|-------|-----|
| - | 001000xxb | 02h | 64 位 | 16 位 | - |

请求参数与数据包括:

- UID 参数

当处于 QUIET 状态时:

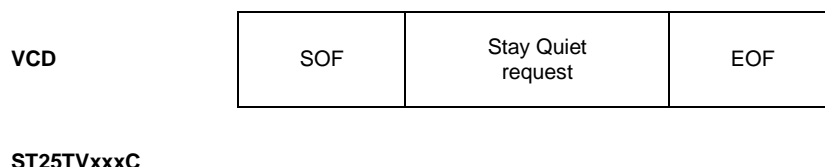
- 如果 Inventory_flag 设为 1, 则 ST25TVxxxC 不处理任何请求,
- ST25TVxxxC 只处理 Address_flag 设为 1 的请求。

ST25TVxxxC 在以下情况下退出 QUIET 状态:

- 被重置 (掉电)。

- 在成功的 Select 请求后，进入 SELECTED 状态。
- 在成功的 ResetToReady 请求后，进入 READY 状态。

图 14. Stay Quiet 帧



6.4.3 ReadSingleBlock

在收到 ReadSingleBlock 请求时，ST25TVxxxC 读取被请求的块，并在响应中发回其 32 位值。

仅在请求块可用并被授予读访问权限时（即，父区域不受读保护，或安全会话打开），ReadSingleBlock 指令适用且会成功。

当 Option_flag 设为 1 时，请求块的块安全状态包含在响应中。

表 102. ReadSingleBlock 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | Block_number | CRC_B | EOF |
|-----|---------------|-----|--------------------|--------------|-------|-----|
| - | 0xxx00xxb | 20h | 64 位 | 8 位 | 16 位 | - |

1. 当 Request_flags=0x1000xxb 时，UID 字段存在

请求参数与数据包括：

- Address_flag 设为 1 时的 UID 参数
- 采用 1 字节编码的 Block_number

表 103. Error_flag 等于 0 时的 ReadSingleBlock 响应格式

| SOF | Response_flags | BSS ⁽¹⁾ | 数据 | CRC_B | EOF |
|-----|----------------|--------------------|--------|-------|-----|
| - | 00h | 8 位 | [32 位] | 16 位 | - |

1. 当 Request_flags=01xx00xxb 时，BSS 字段存在。

当 Error_flag 设为 0 时，响应数据包括：

- Option_flag 设为 1 时的块安全状态（参见表 31. 块安全状态）
- 四字节块数据

提示

表 103 的数据字段可能受 ANDEF 特性影响（参见第 5.4.2 节 增强型 NDEF 描述）

当 Error_flag 设为 1 时，Error_code 字段可在 ReadSingleBlock 响应中取表 104 的值。

表 104. Error_flag 等于 1 时的 ReadSingleBlock 错误代码

| 错误代码 | 说明 |
|------|----------------------|
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 10h | 请求的块不可用 |
| 15h | 对请求的块的读访问受到保护，安全会话关闭 |

当 VICC 响应 ReadSingleBlock 请求时，帧交换的时序为图 11. VCD 与 ST25TVxxxC 之间的读之类帧交换中所述的读之类指令的时序。

6.4.4 WriteSingleBlock

收到 WriteSingleBlock 请求时，ST25TVxxxC 将请求中所包含的数据写入目标块中，并在响应中报告写操作是否成功。

仅在请求块可用并被授予写访问权限时（即，块未锁定、区域不受写保护，或安全会话打开），WriteSingleBlock 指令适用且会成功。

当 Option_flag 设为 1 时，响应被推迟到后续 EOF 请求。

表 105. WriteSingleBlock 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | Block_number | 数据 | CRC_B | EOF |
|-----|---------------|-----|--------------------|--------------|--------|-------|-----|
| - | 0xxx00xxb | 21h | 64 位 | 8 位 | [32 位] | 16 位 | - |

1. 当 Request_flags=0x1000xxb 时，UID 字段存在

请求参数与数据包括：

- Address_flag 设为 1 时的 UID 参数
- 采用 1 字节编码的 Block_number
- 四字节块数据

提示

表 105 的数据字段不受 ANDEF 特性影响（参见第 5.4.2 节 增强型 NDEF 描述）

表 106. Error_flag 等于 0 时的 WriteSingleBlock 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

当 Error_flag 设为 0 时，Response_flags 和 CRC_B 字段之间未插入数据。

当 Error_flag 设为 1 时，Error_code 字段可在 WriteSingleBlock 响应中取表 107 的值。

表 107. Error_flag 等于 1 时的 WriteSingleBlock 错误代码

| 错误代码 | 说明 |
|------|----------------------|
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 10h | 请求的块不可用 |
| 12h | 对请求的块的写访问受到保护，安全会话关闭 |
| 13h | 请求块的编程失败 |

当 VICC 响应 WriteSingleBlock 请求时，帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 Wt 期间，不得进行任何调制（无论是 100%，还是 10%），否则 ST25TVxxxC 可能无法正确地将数据编程到存储器中。

6.4.5 LockBlock

收到 LockBlock 请求时，ST25TVxxxC 永久锁定相应的块值，避免向其内容中写入新值。

仅在请求块可用并被授予写访问权限时（即，块未锁定、区域不受写保护，或安全会话打开），LockBlock 指令适用且会成功。

当 Option_flag 设为 1 时，响应被推迟到后续 EOF 请求。

表 108. LockBlock 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | Block_number | CRC_B | EOF |
|-----|---------------|-----|--------------------|--------------|-------|-----|
| - | 0xxx00xxb | 22h | 64 位 | 8 位 | 16 位 | - |

1. *Request_flags* = 0x1000xxb 时 *UID* 字段存在

请求参数与数据包括：

- *Address_flag* 设为 1 时的 *UID* 参数
- 采用 1 字节编码的 *Block_number*

表 109. *Error_flag* 等于 0 时的 **LockBlock** 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

当 *Error_flag* 设为 0 时，*Response_flags* 和 *CRC_B* 字段之间未插入数据。

当 *Error_flag* 设为 1 时，*Error_code* 字段可在 **LockBlock** 响应中取表 110 的值。

表 110. *Error_flag* 等于 1 时的 **LockBlock** 错误代码

| 错误代码 | 说明 |
|------|---------------------------|
| 02h | 无效请求格式 |
| 03h | 无效 <i>Request_flags</i> 值 |
| 10h | 请求的块不可用 |
| 11h | 请求的块已锁定 |
| 12h | 对请求的块的写访问受到保护，安全会话关闭 |
| 14h | 锁定请求块失败 |

当 **VICC** 响应请求时，帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 *Wt* 期间，不得进行任何调制（无论是 100%，还是 10%），否则 **ST25TVxxxC** 可能无法正确地将 **LCK_BLOCK** 寄存器编程到存储器中。

6.4.6

ReadMultipleBlocks

收到 **ReadMultipleBlocks** 请求时，**ST25TVxxxC** 读取选定块，并在响应中以 32 位的倍数发回其值。

仅在请求的第一个块可用并被授予读访问权限时（即，父区域不受读保护，或安全会话打开），**ReadMultipleBlocks** 指令适用且会成功。

当请求的块范围在用户存储器之外或在无读访问权限的区域结束时，用于响应数据的块范围会在不可用/不可读的块之前被截断。

当 *Option_flag* 设为 1 时，块读取的块安全状态包含在响应中。

表 111. **ReadMultipleBlocks** 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | Block_number | Additional_blocks | CRC_B | EOF |
|-----|---------------|-----|--------------------|--------------|-------------------|-------|-----|
| - | 0xxx00xxb | 23h | 64 位 | 8 位 | 8 位 | 16 位 | - |

1. *Request_flags* = 0x1000xxb 时，*UID* 字段存在

请求参数与数据包括：

- *Address_flag* 设为 1 时的 *UID* 参数
- 采用 1 字节编码的 *Block_number*，以 *Block_number* 开始的请求块范围
- 采用 1 字节编码的 *Additional_blocks*，以 *Block_number* + *Additional_blocks* 结束的请求块范围

表 112. Error_flag 等于 0 时的 ReadMultipleBlocks 响应格式

| SOF | Response_flags | BSS ⁽¹⁾ | 数据 | CRC_B | EOF |
|-----|----------------|--------------------|-----------------------|-------|-----|
| - | 00h | 8 位 ⁽²⁾ | [32 位] ⁽²⁾ | 16 位 | - |

1. 当 Request_flags=01xx00xxb 时, BSS 字段存在

2. 按需要重复

当 Error_flag 设为 0 时, 每个块的响应数据包括:

- Option_flag 设为 1 时的块安全状态 (参见表 31. 块安全状态)
- 四字节块数据

提示

表 112 的数据字段可能受 ANDEF 特性影响 (参见第 5.4.2 节 增强型 NDEF 描述)

当 Error_flag 设为 1 时, Error_code 字段可在 ReadMultipleBlocks 响应中取表 113 的值。

表 113. Error_flag 等于 1 时的 ReadMultipleBlocks 错误代码

| 错误代码 | 说明 |
|------|-----------------------|
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 10h | 请求的块不可用 |
| 15h | 对请求的块的读访问受到保护, 安全会话关闭 |

当 VICC 响应 ReadMultipleBlocks 请求时, 帧交换的时序为图 11. VCD 与 ST25TVxxxC 之间的读之类帧交换中所读的读之类指令的时序。

6.4.7

Select

收到 Select 请求时:

- 若 UID 参数与自己的 UID 匹配, 则 ST25TVxxxC 进入或保持选中状态并发送响应。
- 若 UID 参数与自己的 UID 不匹配, 则已选定的 ST25TVxxxC 将返回就绪状态, 并且不会发送响应。
- 如果发生错误, ST25TVxxxC 将保持其当前状态。

Select_flag 设为 0, Address_flag 设为 1: Select 请求必须在寻址模式下发出。

Option_flag 设为 0: 没有支持的选项。

表 114. Select 请求格式

| SOF | Request_flags | 操作码 | UID | CRC_B | EOF |
|-----|---------------|-----|------|-------|-----|
| - | 001000xxb | 25h | 64 位 | 16 位 | - |

请求参数与数据包括:

- UID 参数

表 115. Error_flag 等于 0 时的选择响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

- 当 Error_flag 设为 0 时, Response_flags 和 CRC_B 字段之间未插入数据。
- 当 Error_flag 设为 1 时, Error_code 字段可在 Select 响应中取表 116 的值。

表 116. Error_flag 等于 1 时的 Select 错误代码

| 错误代码 | 说明 |
|------|--------------------|
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |

当 VICC 响应 Select 请求时，帧交换的时序为图 11. VCD 与 ST25TVxxxC 之间的读之类帧交换中所述的读之类指令的时序。

6.4.8

ResetToReady

收到 ResetToReady 请求时：

- 如果没有发生错误，ST25TVxxxC 将进入或保持 READY 状态。
- 在 SELECTED 状态下，使用寻址模式时，ST25TVxxxC 响应错误
- 在 QUIET 状态下，即使使用非寻址模式，ST25TVxxxC 也处理请求。
- 如果发生错误，ST25TVxxxC 将保持其当前状态。

Option_flag 设为 0：没有支持的选项。

表 117. ResetToReady 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | CRC_B | EOF |
|-----|---------------|-----|--------------------|-------|-----|
| - | 00xx00xxb | 26h | 64 位 | 16 位 | - |

1. Request_flags=0x1000xxb 时，UID 字段存在

请求参数与数据包括：

- Address_flag 设为 1 时的 UID 参数

表 118. Error_flag 等于 0 时的 ResetToReady 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

当 Error_flag 设为 0 时，Response_flags 和 CRC_B 字段之间未插入数据。

当 Error_flag 设为 1 时，Error_code 字段可在 ResetToReady 响应中取表 119 的值。

表 119. Error_flag 等于 1 时的 ResetToReady 错误代码

| 错误代码 | 说明 |
|------|--------------------|
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |

当 VICC 响应 ResetToReady 请求时，帧交换的时序为图 11. VCD 与 ST25TVxxxC 之间的读之类帧交换中所述的读之类指令的时序。

6.4.9

WriteAFI

收到 WriteAFI 请求时，ST25TVxxxC 对 8 位 AFI 寄存器进行编程。

仅在允许 WriteAFI 指令的情况下（即，AFI 未锁定，AFI_PROT=0b 或 AREA1 安全会话打开），WriteAFI 指令才适用且会成功。

当 Option_flag 设为 1 时，响应被推迟到后续 EOF 请求。

表 120. WriteAFI 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | AFI | CRC_B | EOF |
|-----|---------------|-----|--------------------|-----|-------|-----|
| - | 0xxx00xxb | 27h | 64 位 | 8 位 | 16 位 | - |

1. 当 *Request_flags*=0x1000xxb 时, *UID* 字段存在

请求参数与数据包括:

- *Address_flag* 设为 1 时的 *UID* 参数
- 采用 1 字节编码的 *AFI* 参数, 用于对 *AFI* 寄存器进行编程

表 121. *Error_flag* 等于 0 时的 WriteAFI 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

当 *Error_flag* 设为 0 时, *Response_flags* 和 *CRC_B* 字段之间未插入数据。

当 *Error_flag* 设为 1 时, *Error_code* 字段可在 WriteAFI 响应中取表 122 的值。

表 122. *Error_flag* 等于 1 时的 WriteAFI 错误代码

| 错误代码 | 说明 |
|------|---|
| 02h | 无效请求格式 |
| 03h | 无效 <i>Request_flags</i> 值 |
| 12h | <i>LCK_AFI</i> =1b 或 (<i>AFI_PROT</i> =1b 且 <i>AREA1</i> 安全会话关闭) |
| 13h | <i>AFI</i> 寄存器编程失败 |

当 VICC 响应 WriteAFI 请求时, 帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 *Wt* 期间, 不得进行任何调制 (无论是 100%, 还是 10%), 否则 ST25TVxxxC 可能无法正确地将 *AFI* 寄存器编程到存储器中。

6.4.10

LockAFI

收到 LockAFI 请求时, ST25TVxxxC 永久锁定 *AFI* 寄存器。

仅在允许 LockAFI 指令的情况下 (即, *AFI* 未锁定, *AFI_PROT*=0b 或 *AREA1* 安全会话打开), LockAFI 指令才适用且会成功。

当 *Option_flag* 设为 1 时, 响应被推迟到后续 EOF 请求。

表 123. LockAFI 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | CRC_B | EOF |
|-----|---------------|-----|--------------------|-------|-----|
| - | 0xxx00xxb | 28h | 64 位 | 16 位 | - |

1. 当 *Request_flags*=0x1000xxb 时, *UID* 字段存在

请求参数与数据包括:

- *Address_flag* 设为 1 时的 *UID* 参数

表 124. *Error_flag* 等于 0 时的 LockAFI 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

当 *Error_flag* 设为 0 时, *Response_flags* 和 *CRC_B* 字段之间未插入数据。

当 **Error_flag** 设为 1 时，**Error_code** 字段可在 **LockAFI** 响应中取表 125 的值。

表 125. **Error_flag** 等于 1 时的 **LockAFI** 错误代码

| 错误代码 | 说明 |
|------|--|
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 11h | LCK_AFI=1b : LockAFI 指令已成功执行 |
| 12h | AFI_PROT=1b 和 AREA1 安全会话已关闭 |
| 14h | LCK_AFI 寄存器编程失败 |

当 **VICC** 响应 **LockAFI** 请求时，帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 **Wt** 期间，不得进行任何调制（无论是 100%，还是 10%），否则 **ST25TVxxxC** 可能无法正确地将 **LCK_AFI** 寄存器编程到存储器中。

6.4.11

WriteDSFID

收到 **WriteDSFID** 请求时，**ST25TVxxxC** 对 8 位 **DSFID** 寄存器进行编程。仅当 **DSFID** 寄存器未锁定时（**LCK_DSFI**D=0b），**WriteDSFID** 指令才适用且会成功。

当 **Option_flag** 设为 1 时，响应被推迟到后续 **EOF** 请求。

表 126. **WriteDSFID** 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | DSFID | CRC_B | EOF |
|-----|----------------------|-----|--------------------|--------------|--------------|-----|
| - | 0xxx00xxb | 29h | 64 位 | 8 位 | 16 位 | - |

1. 当 **Request_flags=0x1000xxb** 时，**UID** 字段存在

请求参数与数据包括：

- **Address_flag** 设为 1 时的 **UID** 参数。
- 采用 1 字节编码的 **DSFID** 参数，用于对 **DSFID** 寄存器进行编程

表 127. **Error_flag** 等于 0 时的 **WriteDSFID** 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|-----------------------|--------------|-----|
| - | 00h | 16 位 | - |

当 **Error_flag** 设为 0 时，**Response_flags** 和 **CRC_B** 字段之间未插入数据。

当 **Error_flag** 设为 1 时，**Error_code** 字段可在 **WriteDSFID** 响应中取表 128 的值。

表 128. **Error_flag** 等于 1 时的 **WriteDSFID** 错误代码

| 错误代码 | 说明 |
|------|--|
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 12h | LCK_DSFI D=1b: DSFID 寄存器锁定 |
| 13h | DSFID 寄存器编程失败 |

当 **VICC** 响应 **WriteDSFID** 请求时，帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 **Wt** 期间，不得进行任何调制（无论是 100%，还是 10%），否则 **ST25TVxxxC** 可能无法正确地将 **DSFID** 寄存器编程到存储器中。

6.4.12

LockDSFID

收到 LockDSFID 请求时，ST25TVxxxC 永久锁定 DSFID 寄存器。仅当 DSFID 寄存器未锁定时（LCK_DSFIID=0b），LockDSFID 指令才适用且会成功。

当 Option_flag 设为 1 时，响应被推迟到后续 EOF 请求。

表 129. LockDSFID 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | CRC_B | EOF |
|-----|---------------|-----|--------------------|-------|-----|
| - | 0xxx00xxb | 2Ah | 64 位 | 16 位 | - |

1. Request_flags=0x1000xxb 时，UID 字段存在

请求参数与数据包括：

- Address_flag 设为 1 时的 UID 参数

表 130. Error_flag 等于 0 时的 LockDSFID 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

当 Error_flag 设为 0 时，Response_flags 和 CRC_B 字段之间未插入数据。

当 Error_flag 设为 1 时，Error_code 字段可在 LockDSFID 响应中取表 131 的值。

表 131. Error_flag 等于 1 时的 LockDSFID 错误代码

| 错误代码 | 说明 |
|------|----------------------------------|
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 11h | LCK_DSFIID=1b: LockDSFID 指令已成功执行 |
| 14h | LCK_DSFIID 寄存器编程失败 |

当 VICC 响应 LockDSFID 请求时，帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 Wt 期间，不得进行任何调制（无论是 100%，还是 10%），否则 ST25TVxxxC 可能无法正确地将 LCK_DSFIID 寄存器编程到存储器中。

6.4.13

GetSystemInfo

收到 GetSystemInfo 请求时，ST25TVxxxC 会在响应中发回其信息数据。

Option_flag 设为 0：没有支持的选项。

表 132. GetSystemInfo 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | CRC_B | EOF |
|-----|---------------|-----|--------------------|-------|-----|
| - | 00xx00xxb | 2Bh | 64 位 | 16 位 | - |

1. Request_flags=0x1000xxb 时，UID 字段存在

请求参数与数据包括：

- Address_flag 设为 1 时的 UID 参数

表 133. GetSystemInfo 响应

| SOF | Response_flags | Information_flags | UID | DSFID | AFI | Memory_size | IC_ref | CRC_B | EOF |
|-----|----------------|-------------------|------|-------|-----|-------------|--------|-------|-----|
| - | 00h | 0Fh | 64 位 | 8 位 | 8 位 | 16 位 | 08h | 16 位 | - |

当 Error_flag 设为 0 时，响应数据包括：

- 采用 1 字节编码的 Information_flags，设为 0Fh（DSFID、AFI、Memory_size 和 IC_ref fields 都存在）。
- UID 寄存器值
- DSFID 寄存器值
- AFI 寄存器值
- 采用 2 字节编码的 Memory_size：
 - 8-MSB (03h) = 块大小（字节数）- 1
 - 8-LSB (END_MEM) = 用户存储器容量（块数）- 1
- IC_REF 寄存器值

当 Error_flag 设为 1 时，Error_code 字段可在 GetSystemInfo 响应中取表 134 的值。

表 134. Error_flag 等于 1 时的 GetSystemInfo 错误代码

| 错误代码 | 说明 |
|------|--------------------|
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |

当 VICC 响应 GetSystemInfo 请求时，帧交换的时序为图 11 中所述的读之类指令的时序。

6.4.14

GetMultipleBlockSecurityStatus

收到 GetMultipleBlockSecurityStatus 请求时，ST25TVxxxC 以所选块的块安全状态响应。

仅在请求的第一个块可用时，GetMultipleBlockSecurityStatus 指令才适用且会成功。

当请求的块范围在用户存储器之外结束时，用于响应数据的块范围会在最后一个可用的块处被截断。

Option_flag 设为 0：没有支持的选项。

表 135. GetMultipleBlockSecurityStatus 请求格式

| SOF | Request_flags | 操作码 | UID ⁽¹⁾ | Block_number | Additional_blocks | CRC_B | EOF |
|-----|---------------|-----|--------------------|--------------|-------------------|-------|-----|
| - | 00xx00xxb | 2Ch | 64 位 | 8 位 | 8 位 | 16 位 | - |

1. 当 Request_flags=0x1000xxb 时，UID 字段存在

请求参数与数据包括：

- Address_flag 设为 1 时的 UID 参数
- 采用 1 字节编码的 Block_number，以 Block_number 开始的请求块范围
- 采用 1 字节编码的 Additional_blocks，以 Block_number + Additional_blocks 结束的请求块范围

表 136. Error_flag 等于 0 时的 GetMultipleBlockSecurityStatus 响应格式

| SOF | Response_flags | BSS | CRC_B | EOF |
|-----|----------------|--------------------|-------|-----|
| - | 00h | 8 位 ⁽¹⁾ | 16 位 | - |

1. 按需要重复

当 Error_flag 设为 0 时，每个块的响应数据包括：

- 块安全状态（参见表 31. 块安全状态）

当 Error_flag 设为 1 时，Error_code 字段可在 GetMultipleBlockSecurityStatus 响应中取表 137 的值。

表 137. Error_flag 等于 1 时的 GetMultipleBlockSecurityStatus 错误代码

| 错误代码 | 说明 |
|------|--------------------|
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 10h | 请求的块不可用 |

当 VICC 响应 GetMultipleBlockSecurityStatus 请求时，帧交换的时序为图 11 中所述的读之类指令的时序。

6.4.15

ExtendedGetSystemInfo

收到 ExtendedGetSystemInfo 请求时，ST25TVxxxC 会在响应中发回其信息数据。

Option_flag 设为 0：没有支持的选项。

表 138. ExtendedGetSystemInfo 请求格式

| SOF | Request_flags | 操作码 | Information_request_list | UID ⁽¹⁾ | CRC_B | EOF |
|-----|---------------|-----|--------------------------|--------------------|-------|-----|
| - | 00xx00xxb | 3Bh | 0xx1xxxxb | 64 位 | 16 位 | - |

1. 当 Request_flags=0x1000xxb 时，UID 字段存在

请求参数与数据包括：

- 采用 1 字节编码的 Information_request_list 参数，参见下面的表 139
- Address_flag 设为 1 时的 UID 参数

表 139. Information_request_list 内容

| 位 | 请求的信息 | 说明 |
|----|--------------|--|
| b0 | DSFID | 0: 未请求 DSFID 1: 已请求 DSFID |
| b1 | AFI | 0: 未请求 AFI 1: 已请求 AFI |
| b2 | Memory_size | 0: 未请求 VICC 存储器容量 1: 已请求 VICC 存储器容量 |
| b3 | IC_ref | 0: 未请求 IC 参数 1: 已请求 IC 参数 |
| b4 | MOI | 1: 始终在响应标志中返回 MOI 信息 |
| b5 | Command_list | 0: 未请求支持的指令列表 1: 已请求支持的指令列表 |
| b6 | CSI 信息 | 0: 未请求 CSI 列表 1: 已请求 CSI 列表 |
| b7 | Ext_list | 0: Information_request_list 的大小为 1 字节 |

表 140. Error_flag 等于 0 时的 ExtendedGetSystemInfo 响应格式

| SOF | Response_flags | Information_flags | UID | Information_fields | CRC_B | EOF |
|-----|----------------|-------------------|------|--------------------|-------|-----|
| - | 00h | 00x0xxxxb | 64 位 | 高达 80 位 | 16 位 | - |

表 141. Information_flags 内容

| 位 | 响应信息 | 说明 |
|----|-----------------|---|
| b0 | DSFID | 0: Information_fields 中不存在 DSFID 1: Information_fields 中存在 DSFID |
| b1 | AFI | 0: Information_fields 中不存在 AFI 1: Information_fields 中存在 AFI |
| b2 | Memory_size | 0: Information_fields 中不存在 Memory_size 1: Information_fields 中存在 Memory_size |
| b3 | IC_ref | 0: Information_fields 中不存在 IC_ref 1: Information_fields 中存在 IC_ref |
| b4 | MOI | 0: 1 字节寻址 |
| b5 | Command_list | 0: Information_fields 中不存在 Command_list 1: Information_fields 中存在 Command_list |
| b6 | CSI_information | 0: CSI 列表不存在 |
| b7 | Ext_info | 0: Information_flags 的大小为 1 字节 |

表 142. Information_fields 内容

| DSFID ⁽¹⁾ | AFI ⁽¹⁾ | Memory_size ⁽¹⁾ | IC_ref ⁽¹⁾ | Command_list ⁽¹⁾ |
|----------------------|--------------------|----------------------------|-----------------------|-----------------------------|
| 8 位 | 8 位 | 24 位 | 08h | 00003FEh |

1. 信息字段存在与否取决于 *Information_flags* 的值

当 *Error_flag* 设为 0 时，响应数据包括：

- 采用 1 字节编码的 *Information_flags*，定义了存在哪些字段（参见表 141）
- UID 寄存器值
- DSFID 寄存器值，*Information_flags*[0]=1b 时存在
- AFI 寄存器值，*Information_flags*[1]=1b 时存在
- 采用 3 字节编码的 VICC 存储器容量，*Information_flags*[2]=1b 时存在
 - 8-MSB (03h) = 块大小（字节数）- 1
 - 16-LSB (END_MEM) = 用户存储器容量（块数）- 1
- IC_REF 寄存器值，*Information_flags*[3]=1b 时存在
- 采用 4 字节编码的 VICC 指令列表，*Information_flags*[5]=1b 时存在

当 *Error_flag* 设为 1 时，*Error_code* 字段可在 *ExtendedGetSystemInfo* 响应中取表 143 的值。

表 143. *Error_flag* 等于 1 时的 *ExtendedGetSystemInfo* 错误代码

| 错误代码 | 说明 |
|------|---|
| 02h | 无效请求格式 |
| 03h | 无效 <i>Request_flags</i> 或 <i>Information_request_list</i> 值 |

当 VICC 响应 *ExtendedGetSystemInfo* 请求时，帧交换的时序为图 11 中所述的读之类指令的时序。

6.4.16

ReadConfiguration

收到 *ReadConfiguration* 请求时，ST25TVxxxC 读取所选配置寄存器，并在响应中返回其值。

仅在请求的配置寄存器（通过 FID/PID 对来识别）可用且已被授予读访问权限时（即不受读保护、功能未锁定且 CONFIG 安全会话打开），*ReadConfiguration* 指令才适用且会成功。

Option_flag 设为 0：没有支持的选项。

表 144. ReadConfiguration 请求格式

| SOF | Request_flags | 操作码 | IC Mfg 编码 | UID ⁽¹⁾ | FID | PID | CRC_B | EOF |
|-----|---------------|-----|-----------|--------------------|-----|-----|-------|-----|
| - | 00xx00xxb | A0h | 02h | 64 位 | 8 位 | 8 位 | 16 位 | - |

1. 当 *Request_flags*=0x1000xxb 时, *UID* 字段存在

请求参数与数据包括:

- 采用 1 字节编码的 IC 厂商码, 值应为 02h
- *Address_flag* 设为 1 时的 *UID* 参数
- 采用 1 字节编码的 *FID* 参数
- 采用 1 字节编码的 *PID* 参数

表 145. *Error_flag* 等于 0 时的 ReadConfiguration 响应格式

| SOF | Response_flags | Data 请求的 FID 与 PID 值。 ⁽¹⁾ | CRC_B | EOF |
|-----|----------------|--------------------------------------|-------|-----|
| - | 00h | 8 至 64 位 | 16 位 | - |

1. 响应的数据大小取决于根据

2. 表 4. 配置寄存器列表

当 *Error_flag* 设为 0 时, 响应数据包括:

- 采用 1 到 8 字节编码 (取决于请求的 *FID*/*PID* 对) 的配置寄存器值 (参见表 4. 配置寄存器列表)

提示

当寄存器值采用多字节编码时, 在响应 *ReadConfiguration* 请求时, 以 *LSB* 到 *MSB* 字节顺序传输该值。

当 *Error_flag* 设为 1 时, *Error_code* 字段可在 *ReadConfiguration* 响应中取表 146 的值。

表 146. *Error_flag* 等于 1 时的 ReadConfiguration 错误代码

| 错误代码 | 说明 |
|------|---|
| 01h | 无效 IC Mfg 代码值 |
| 02h | 无效请求格式 |
| 03h | 无效 <i>Request_flags</i> 值 |
| 10h | 请求的 <i>FID</i> / <i>PID</i> 不可用 |
| 15h | 对请求的 <i>FID</i> / <i>PID</i> 的读访问受到保护, <i>CONFIG</i> 安全会话关闭 |

当 *VICC* 响应 *ReadConfiguration* 请求时, 帧交换的时序为图 11 中所述的读之类指令的时序。

6.4.17

WriteConfiguration

收到 *WriteConfiguration* 请求时, ST25TVxxxC 将请求中所包含的数据写入所选的配置寄存器中, 如果写操作成功, 则已确认回应。

仅在请求的配置寄存器 (通过 *FID*/*PID* 对来识别) 可用且已被授予写访问权限时 (即功能未锁定且 *CONFIG* 安全会话打开), *WriteConfiguration* 指令才适用且会成功。

当 *Option_flag* 设为 1 时, 响应被推迟到后续 *EOF* 请求。

表 147. WriteConfiguration 请求格式

| SOF | Request_flags | 操作码 | IC Mfg 编码 | UID ⁽¹⁾ | FID | PID | 数据 | CRC_B | EOF |
|-----|---------------|-----|-----------|--------------------|-----|-----|--------|-------|-----|
| - | 0xxx00xxb | A1h | 02h | 64 位 | 8 位 | 8 位 | 8-32 位 | 16 位 | - |

1. 当 *Request_flags*=0x1000xxb 时, *UID* 字段存在

请求参数与数据包括:

- 采用 1 字节编码的 IC 厂商码, 值应为 02h

- Address_flag 设为 1 时的 UID 参数
- 采用 1 字节编码的 FID 参数
- 采用 1 字节编码的 PID 参数
- 采用 1 到 4 字节编码（取决于请求的 FID/PID 对）的新寄存器值（参见表 4. 配置寄存器列表）

提示

当寄存器值采用多字节编码时，在 *WriteConfiguration* 请求中，以 *LSB* 到 *MSB* 字节顺序传输该值。

表 148. Error_flag 等于 0 时的 *WriteConfiguration* 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

Error_flag 设为 0 时：

- 在 Response_flags 与 CRC_B 字段之间未插入数据。
- 成功将寄存器值更新到存储器中，新值通过 *ReadConfiguration* 请求立即可读。但是，根据所选配置，新值可能会立即生效，或在下一个 RF 启动序列生效（参见表 4. 配置寄存器列表的激活时间列）。

当新配置寄存器值在下一个 RF 启动序列生效时，先前的配置值的效果一直持续到将寄存器更新到存储器之后，直至 ST25TVxxxC 进入 POWER-OFF 状态。

当 Error_flag 设为 1 时，Error_code 字段可在 *WriteConfiguration* 响应中取表 149 的值。

表 149. Error_flag 等于 1 时的 *WriteConfiguration* 错误代码

| 错误代码 | 说明 |
|------|--------------------------------------|
| 01h | 无效 IC Mfg 代码值 |
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 10h | 请求的 FID/PID 不可用 |
| 11h | LCK_CONFIG (FID=FFh,PID=00h)的位已设为 1b |
| 12h | 对请求的 FID/PID 的写+访问受到保护，CONFIG 安全会话关闭 |
| 13h | 对请求的 FID/PID（不同于 LCK_CONFIG）的编程失败 |
| 14h | LCK_CONFIG (FID=FFh,PID=00h)编程失败 |

当 VICC 响应 *WriteConfiguration* 请求时，帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 *Wt* 期间，不得进行任何调制（无论是 100%，还是 10%），否则 ST25TVxxxC 可能无法正确地将配置寄存器编程到存储器中。

6.4.18

WritePassword

收到 *WritePassword* 请求时，ST25TVxxxC 使用请求中所包含的数据来修改所选密码，如果写操作成功，则已确认回应。

仅在成功执行 *PresentPassword* 指令并选择相同的密码时，*WritePassword* 指令才适用且会成功。有关密码管理的详细信息，请参见第 5.1.2 节 密码管理。

当 Option_flag 设为 1 时，响应被推迟到后续 EOF 请求。

表 150. WritePassword 请求格式

| SOF | Request_flags | 操作码 | IC Mfg 编码 | UID ⁽¹⁾ | Password_id | Password_data | CRC_B | EOF |
|-----|---------------|-----|-----------|--------------------|-------------|---------------|-------|-----|
| - | 0xxx00xxb | B1h | 02h | 64 位 | 8 位 | 32 或者 64 位 | 16 位 | - |

1. 当 Request_flags=0x1000xxb 时，UID 字段存在

请求参数与数据包括：

- 采用 1 字节编码的 IC 厂商码，值应为 02h
- Address_flag 设为 1 时的 UID 参数
- 采用 1 字节编码的 Password_id
- 根据采用 4 或 8 字节编码的 Password_data 数据表 151. 密码寄存器列表

通过如第 5.1.3 节 密码加密中所述加密密码的新简单值来获得 Password_data 值。

提示 如果将简单值误用于 WritePassword 指令的 Password_data 字段，则在 ST25TVxxxC 器件上用 PresentPassword 指令提供其加密值将失败。

提示 ST25TVxxx 与 ST25TVxxxC 器件在 Password_data 字段加密方面的 WritePassword 指令行为不同。Password_data 字段为 DS12074 数据手册中所述的 ST25TVxxx 器件的简单密码值，同时也是本文档中所述的 ST25TVxxxC 器件上的加密密码值。

建议在寻址或选择模式下发出 WritePassword 请求，以提高系统的稳健性。

这样可确保密码更改只应用于特定标签/UID。

表 151. 密码寄存器列表

| 名称 | Password_id | Password_data 大小 |
|----------|-------------|---------------------------|
| PWD_CFG | 00h | 4 字节 |
| PWD_A1 | 01h | 如果 END_A1<END_MEM，则为 4 字节 |
| | | 如果 END_A1=END_MEM，则为 8 字节 |
| PWD_A2 | 02h | 如果 END_A1<END_MEM，则为 4 字节 |
| | | 如果 END_A1=END_MEM，则为无效请求 |
| PWD_UNTR | 03h | 4 字节 |

表 152. Error_flag 等于 0 时的 WritePassword 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

当 Error_flag 设为 0 时：

- 在 Response_flags 与 CRC_B 字段之间未插入数据。
- 密码已成功更新到存储器中，相应的安全会话保持打开。

当 Error_flag 设为 1 时，Error_code 字段可在 WritePassword 响应中取表 153 的值。

表 153. Error_flag 等于 1 时的 WritePassword 错误代码

| 错误代码 | 说明 |
|------|--------------------|
| 01h | 无效 IC Mfg 代码值 |
| 02h | 无效请求格式，包括密码大小无效的情况 |
| 03h | 无效 Request_flags 值 |
| 10h | 无效 Password_id 值 |
| 12h | 安全会话已关闭 |
| 13h | 请求的密码编程失败 |

当 VICC 响应 WritePassword 请求时，帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 Wt 期间，不得进行任何调制（无论是 100%，还是 10%），否则 ST25TVxxxC 可能无法正确地将密码值编程到存储器中。

在操作 WritePassword 指令期间，无防撕裂机制。指令应在具有稳定 RF 场的情况下进行，否则写操作可能无法正确完成，并可能意味着密码内容的损失/损坏。

当此类内容损失/损坏发生时，ST25TVxxxC 提供密码恢复功能，参见第 5.1.2 节 密码管理。

6.4.19

PresentPassword

收到 PresentPassword 请求时，ST25TVxxxC 将所选的密码寄存器与在请求中编码的密码进行比较，如果操作成功，则以回执回应。

成功执行 PresentPassword 指令后，将打开与密码有关的安全会话，如第 5.1 节 数据保护中所述。

Option_flag 设为 0：没有支持的选项。

表 154. PresentPassword 请求格式

| SOF | Request_flags | 操作码 | IC Mfg 编码 | UID ⁽¹⁾ | Password_id | Password_data | CRC_B | EOF |
|-----|---------------|-----|-----------|--------------------|-------------|---------------|-------|-----|
| - | 00xx00xxb | B3h | 02h | 64 位 | 8 位 | 32 或者 64 位 | 16 位 | - |

1. 当 Request_flags=0x1000xxb 时，UID 字段存在

请求参数与数据包括：

- 采用 1 字节编码的 IC 厂商码，值应为 02h
- Address_flag 设为 1 时的 UID 参数
- 采用 1 字节编码的 Password_id
- 根据采用 4 或 8 字节编码的 Password_data 数据 表 151. 密码寄存器列表

通过第 5.1.3 节 密码加密中所述的简单密码值加密来获得唯一的有效 Password_data 值。

建议在寻址或选择模式下发出 PresentPassword 请求，以提高系统的稳健性。这样可确保密码提供只应用于特定标签/UID。

表 155. Error_flag 等于 0 时的 PresentPassword 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

Error_flag 设为 0 时：

- 在 Response_flags 与 CRC_B 字段之间未插入数据。
- 密码已成功提供，相应的安全会话已打开。

当 Error_flag 设为 1 时，Error_code 字段可在 PresentPassword 响应中取表 156 表 152 中的值。

如果 Password_data 的无效值存在，则所有安全会话将关闭。

ST25TVxxxC 提供密码尝试限制功能，以防止对密码的暴力破解攻击，参见第 5.1.2 节 密码管理。

提示

在通过 PresentPassword / Kill / ToggleUntraceable 指令提供无效的 Password_data 值之后，在尝试通过 PresentPassword 指令再次提供密码之前（如第 5.1.3 节 密码加密中所述），应调用 GetRandomNumber 指令。

表 156. Error_flag 等于 1 时的 PresentPassword 错误代码

| 错误代码 | 说明 |
|------|--------------------|
| 01h | 无效 IC Mfg 代码值 |
| 02h | 无效请求格式，包括密码大小无效的情况 |
| 03h | 无效 Request_flags 值 |
| 0Fh | 无效 Password_data 值 |
| 10h | 无效 Password_id 值 |

当 VICC 响应 PresentPassword 请求时，帧交换的时序为图 11 中所述的读之类指令的时序。

6.4.20

Kill

收到 Kill 请求时，ST25TVxxxC 将 PWD_CFG 寄存器与编码在请求中的密码进行比较，如果操作成功，则已回执回应。

仅当 DIS_KILL 寄存器设为 0b 时，Kill 指令才适用，否则该指令将被忽略。成功执行 Kill 指令后，ST25TVxxxC 永久进入 KILLED 状态，并对任何请求保持静默。

Select_flag 设为 0，Address_flag 设为 1：Kill 请求必须在寻址模式下发出。

当 Option_flag 设为 1 时，响应被推迟到后续 EOF 请求。

表 157. Kill 请求格式

| SOF | Request_flags | 操作码 | IC Mfg 编码 | UID | Password_id | Password_data | CRC_B | EOF |
|-----|---------------|-----|-----------|------|-------------|---------------|-------|-----|
| - | 0x1000xxb | A6h | 02h | 64 位 | 00h | [32 位] | 16 位 | - |

请求参数与数据包括：

- 采用 1 字节编码的 IC 厂商码，值应为 02h
- UID 参数
- 采用 1 字节编码的 Password_id，值应为 00h
- 采用 4 字节编码的 Password_data

通过 中所述的简单密码值加密来获得唯一的有效 Password_data 值 第 5.1.3 节 密码加密

注意

在通过 PresentPassword / Kill / ToggleUntraceable 指令提供无效的 Password_data 值之后，在需要重新提供密码以执行kill指令前，应先调用 GetRandomNumber 指令。

提示

ST25TVxxx 与 ST25TVxxxC 器件在 Password_data 字段加密方面的 Kill 指令行为不同。Password_data 字段为 DS12074 数据手册中所述的 ST25TVxxx 器件是简单密码值而本档中所述的 ST25TVxxxC 器件上为加密密码值。

表 158. Error_flag 等于 0 时的 Kill 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

Error_flag 设为 0 时：

- 在 Response_flags 与 CRC_B 字段之间未插入数据。
- 通过将 KILL_CMD 寄存器设为 1b，ST25TVxxxC 永久进入 KILLED 状态。

当 Error_flag 设为 1 时，Error_code 字段可在 Kill 响应中取表 159 的值。

表 159. Error_flag 等于 1 时的 Kill 错误代码

| 错误代码 | 说明 |
|------|--------------------|
| 01h | 无效 IC Mfg 代码值 |
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 0Fh | 无效 Password_data 值 |
| 10h | 无效 Password_id 值 |
| 14h | KILL_CMD 编程失败 |

当 VICC 响应 Kill 请求时，帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 Wt 期间，不得进行任何调制（无论是 100%，还是 10%），否则 ST25TVxxxC 可能无法正确地将 KILL_CMD 寄存器编程到存储器中。

6.4.21 Initiate

收到 Initiate 请求时，ST25TVxxxC 将 Initiate_flag 寄存器设为 1b 并发回响应。当 ST25TVxxxC 进入 POWER-OFF 状态时，Initiate_flag 被自动重置为 0b。

Select_flag 设为 0，Address_flag 设为 0：Initiate 请求必须在非寻址模式下发出。

Option_flag 设为 0：没有支持的选项。

表 160. Initiate 请求格式

| SOF | Request_flags | 操作码 | IC Mfg 编码 | CRC_B | EOF |
|-----|---------------|-----|-----------|-------|-----|
| - | 000000xxb | D2h | 02h | 16 位 | - |

请求参数与数据包括：

- 采用 1 字节编码的 IC 厂商码，值应为 02h

表 161. Error_flag 等于 0 时的 Initiate 响应格式

| SOF | Response_flags | DSFID | UID | CRC_B | EOF |
|-----|----------------|-------|------|-------|-----|
| - | 00h | 8 位 | 64 位 | 16 位 | - |

当 Error_flag 设为 0，Initiate_flag 设为 1b 时，响应数据包括：

- DSFID 寄存器值
- UID 寄存器值

ST25TVxxxC 在出错时不会产生任何应答。

当 VICC 响应 Initiate 请求时，帧交换的时序为图 11 中所述的读之类指令的时序。

6.4.22 InventoryInitiated

收到 InventoryInitiated 请求时，如果 Initiate_flag 设为 1b，并且参数与 UID 和 AFI 寄存器的值匹配，ST25TVxxxC 发送响应。

Inventory_flag 设为 1：Request_flags 的位 4 和 5 分别对 AFI_flag 和 Nb_slots_flag 进行编码。

Option_flag 设为 0：没有支持的选项。

表 162. InventoryInitiated 请求格式

| SOF | Request_flags | 操作码 | IC Mfg 编码 | AFI ⁽¹⁾ | Mask_length | Mask_value | CRC_B | EOF |
|-----|---------------|-----|-----------|--------------------|-------------|------------|-------|-----|
| - | 00xx01xxb | D1h | 02h | 8 位 | 8 位 | 0-64 位 | 16 位 | - |

1. Request_flags=00x101xxb 时，AFI 字段存在

请求参数与数据包括：

- 采用 1 字节编码的 IC 厂商码，值应为 02h
- AFI_flag 设为 1 时的 AFI 参数
- 以位为单位的 Mask_length 长度，Nb_slots_flag = 0b 时 ≤ 60，Nb_slots_flag = 1b 时 ≤ 64
- Mask_value，以字节为单位的大小为 (Mask_length + 7)/8，如果 Mask_length = 00h，则不存在

表 163. Error_flag 等于 0 时的 InventoryInitiated 响应格式

| SOF | Response_flags | DSFID | UID | CRC_B | EOF |
|-----|----------------|-------|------|-------|-----|
| - | 00h | 8 位 | 64 位 | 16 位 | - |

当 Error_flag 设为 0 时，响应数据包括：

- DSFID 寄存器值

- UID 寄存器值

ST25TVxxxC 在出错时不会产生任何应答。

当 VICC 响应 **InventoryInitiated** 请求时，帧交换的时序为图 11 中所述的读之类指令的时序。

当 **Nb_slots_flag** 设为 0 时，在表 162 中的初始请求后（具有第 6.3 节 时序定义中所述的以下时序），VCD 发出 15 EOF 请求：

- 如果 VICC 响应 EOF 请求，则帧交换的时序为读之类指令的时序
- 如果 VCD 从一个或多个 VICC 收到响应，则在发送下一个 EOF 请求前，其等待时间为 t_2
- 如果 VCD 未从任何 VICC 收到响应，则在发送下一个 EOF 请求前，其等待时间为 t_3

6.4.23

ToggleUntraceable

收到 **ToggleUntraceable** 请求时，ST25TVxxxC 将 **PWD_UNTR** 寄存器与编码在请求中的密码进行比较，如果操作成功，则已确认回应。

ToggleUntraceable 仅在以下情况下适用，否则将被忽略：

- ST25TVxxxC 处于 **READY**、**SELECTED** 或 **QUIET** 状态，请求在寻址模式下发出（**Select_flag**=0，**Address_flag**=1）
- ST25TVxxxC 处于 **UNTRACEABLE** 状态，请求在非寻址模式下发出（**Select_flag**=0，**Address_flag**=0）

在成功执行 **ToggleUntraceable** 指令后，ST25TVxxxC 离开或进入（取决于寻址模式）第 6.2.9 节 自定义状态中所述的 **UNTRACEABLE** 状态。

当 **Option_flag** 设为 1 时，响应被推迟到后续 EOF 请求。

表 164. **ToggleUntraceable** 请求格式

| SOF | Request_flags | 操作码 | IC Mfg 编码 | UID ⁽¹⁾ | Password_id | Password_data | CRC_B | EOF |
|-----|---------------|-----|-----------|--------------------|-------------|---------------|-------|-----|
| - | 0xx000xxb | BAh | 02h | 64 位 | 03h | [32 位] | 16 位 | - |

1. 当 **Request_flags**=0x1000xxb 时，**UID** 字段存在

请求参数与数据包括：

- 采用 1 字节编码的 IC 厂商码，值应为 02h
- **Address_flag** 设为 1 时的 **UID** 参数
- 采用 1 字节编码的 **Password_id**，值应为 03h
- 采用 4 字节编码的 **Password_data**

通过第 5.1.3 节中所述的简单密码值加密来获得唯一的有效 **Password_data** 值

提示

在通过 **PresentPassword / Kill / ToggleUntraceable** 指令提供无效的 **Password_data** 值之后，在尝试通过 **ToggleUntraceable** 指令再次提供密码之前（如第 5.1.3 节 密码加密中所述），应调用 **GetRandomNumber** 指令。

提示

操作码值 **BAh** 用于 **DS12074** 数据手册中所述的 ST25TVxxx 器件的 **EnableUntraceable** 指令。**EnableUntraceable** 指令与 **ToggleUntraceable** 指令具有相同的格式，除了 **Password_id** 字段的值在 ST25TVxxx 器件上为 00h，在 ST25TVxxxC 上为 03h，如本文档所述。

表 165. **Error_flag** 等于 0 时的 **ToggleUntraceable** 响应格式

| SOF | Response_flags | CRC_B | EOF |
|-----|----------------|-------|-----|
| - | 00h | 16 位 | - |

Error_flag 设为 0 时：

- 在 **Response_flags** 与 **CRC_B** 字段之间未插入数据。
- 如果请求在寻址模式下发出，ST25TVxxxC 通过将 **UNTR_CMD** 寄存器设为 1b 进入 **UNTRACEABLE** 状态。
- 如果请求在非寻址模式下发出，ST25TVxxxC 通过将 **UNTR_CMD** 寄存器设为 0b 进入 **READY** 状态。

当 **Error_flag** 设为 1 时，**Error_code** 字段可在 **ToggleUntraceable** 响应中取表 166 的值。

表 166. Error_flag 等于 1 时的 ToggleUntraceable 错误代码

| 错误代码 | 说明 |
|------|--------------------|
| 01h | 无效 IC Mfg 代码值 |
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 0Fh | 无效 Password_data 值 |
| 10h | 无效 Password_id 值 |
| 13h | UNTR_CMD 编程失败 |

当 VICC 响应 ToggleUntraceable 请求时，帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 Wt 期间，不得进行任何调制（无论是 100%，还是 10%），否则 ST25TVxxxC 可能无法正确地将 UNTR_CMD 寄存器编程到存储器中。

6.4.24

GetRandomNumber

收到 GetRandomNumber 请求时，ST25TVxxxC 以 16 位随机数响应。

当 Option_flag 设为 1 时，响应被推迟到后续 EOF 请求。

表 167. GetRandomNumber 请求格式

| SOF | Request_flags | 操作码 | IC Mfg 编码 | UID ⁽¹⁾ | CRC_B | EOF |
|-----|---------------|-----|-----------|--------------------|-------|-----|
| - | 0xxx00xxb | B4h | 02h | 64 位 | 16 位 | - |

1. 当 Request_flags=0x1000xxb 时，UID 字段存在

请求参数与数据包括：

- 采用 1 字节编码的 IC 厂商码，值应为 02h
- Address_flag 设为 1 时的 UID 参数

表 168. Error_flag 等于 0 时的 GetRandomNumber 响应格式

| SOF | Response_flags | RND_NUMBER | CRC_B | EOF |
|-----|----------------|------------|-------|-----|
| - | 00h | 16 位 | 16 位 | - |

当 Error_flag 设为 0 时，新的 16 位值已被编程到 RND_NUMBER 寄存器中，响应数据包括：

- RND_NUMBER 寄存器值

当 Error_flag 设为 1 时，Error_code 字段可在 GetRandomNumber 响应中取表 169 的值。

表 169. Error_flag 等于 1 时的 GetRandomNumber 错误代码

| 错误代码 | 说明 |
|------|--------------------|
| 01h | 无效 IC Mfg 代码值 |
| 02h | 无效请求格式 |
| 03h | 无效 Request_flags 值 |
| 13h | RND_NUMBER 编程失败 |

当 VICC 响应 GetRandomNumber 请求时，帧交换的时序为图 12 和图 13 中所述的写之类指令的时序。

在 RF 写循环 Wt 期间，不得进行任何调制（无论是 100%，还是 10%），否则 ST25TVxxxC 可能无法正确地将 RND_NUMBER 寄存器编程到存储器中。

7 唯一标识符 (UID)

ST25TVxxxC IC 由 64 位唯一标识符 (UID) 唯一标识。此 UID 符合 ISO/IEC 15693 和 ISO/IEC 7816-6 标准。UID 为只读码，它包括：

- 8 字节
- 8 位幻数代码 E0h
- 8 位 IC 厂商码“ST 02h”(ISO/IEC 7816-6/AM1)
- 8 位 ST25TVxxxC 产品代码 08h
- 40 位唯一序列号。

表 170. UID 格式

| MSB | | LSB | |
|---------|---------|--------------|--------|
| b63-b56 | b55-b48 | b47-b40 | b39-b0 |
| E0h | 02h | ST 产品代码: 08h | 唯一序列号。 |

7.1 不可追踪的 UID

当 ST25TVxxxC 满足以下条件之一时：

- 当前 RF 会话以 UNTRACEABLE 状态开始
- 当前状态为 UNTRACEABLE

然后在处理所有指令的请求与响应帧时，UID 寄存器被表 171 中的内容屏蔽，除了在响应 ReadConfiguration 请求时 (FID=FEh, PID=01h)，始终返回 UID 寄存器的内容而不屏蔽。

表 171. 不可追踪的 UID: UNTRACEABLE 状态中的 UID 值

| MSB | | LSB | |
|---------|---------|---------|-------------|
| b63-b56 | b55-b48 | b47-b40 | b39-b0 |
| E0h | 02h | 00h | 0000000000h |

提示

当 VCD 字段中存在多个响应表 171 中的 UID 的 ST25TVxxxC 标签时，无法通过防冲突过程进行区别。为使应用程序正常运行，VCD 字段中仅应存在一个响应不可追踪的 UID 值的 ST25TVxxxC IC

8 设备参数

8.1 最大额定值

如果对设备施加的压力超出了表 172. 绝对最大额定值中列出的额定值，可能会对其造成永久损坏。这些仅仅是耐受额定值，并不意味设备可在这些条件下或是超出本说明书工作原理部分指示的任何条件下工作。设备长时间处在绝对最大额定条件下可能影响设备的可靠性。另请参阅 STMicroelectronics SURE 计划和其他相关的质量文档。

表 172. 绝对最大额定值

| 符号 | 说明 | 最小值 | 最大值 | 单位 |
|-----------------------------------|---|-----|------|----|
| T _A | 环境工作温度 | -40 | 85 | °C |
| T _{STG_1} | UFDFPN5 封装的存储温度 | -65 | 150 | °C |
| T _{STG_2} | 已切割晶圆的存储温度 ⁽¹⁾ | 15 | 25 | °C |
| t _{STG} | 已切割晶圆 ⁽¹⁾ 的存储持续时间以意法半导体的生产日期计数 | - | 9 | 月 |
| V _{MAX_1} ⁽²⁾ | AC0 与 AC1 之间的最大输入电压幅度（峰到峰） | - | 11 | V |
| V _{ESD} | 电子放电电压 ⁽³⁾ | - | 2000 | V |

1. UV 载带上的已切割晶圆以初始封装形式保存

2. (VAC0-VAC1)峰峰值特性测试

3. ANSI/ESDA/JEDEC JS-001 人体模型，在所有引脚上，C = 100 pF、R = 1500 Ω、R2 = 500 Ω

8.2 RF 电气参数

本节概括了器件的工作与测量条件以及 RF 电气参数。

后续 RF 特性表中的参数来自各测量条件下的测试，这些测量条件在相关的表中有概括。当设计人员引用直流和交流特性表中的参数时应检查其所设计电路的测量条件是否与表中描述的工作条件匹配。

表 173. RF 特性

| 符号 | 说明 | 条件 ⁽¹⁾⁽²⁾ | 最小值 | 典型值 | 最大值 | 单位 |
|------------------------|-----------------------------|---|--------|--------|--------|-----|
| f _{CC} | 外部 RF 信号频率 | - | 13.553 | 13.56 | 13.567 | MHz |
| f _{SL} | 低副载波频率（f _{CC} /32） | - | - | 423.75 | - | kHz |
| f _{SH} | 高副载波频率（f _{CC} /28） | - | - | 484.28 | - | kHz |
| MI ₁₀ | 10%载波调制指数 ⁽³⁾ | 150 mA/m < H < 5 A/m | 10 | - | 30 | % |
| MI ₁₀₀ | 100%载波调制指数 | 150 mA/m < H < 5 A/m | 95 | - | 100 | % |
| t _{Boot_RF_1} | RF 启动时间 ⁽⁵⁾ | TD_EVENT_UPDATE_EN=0b , UTC_EN=0b, 来自 H _{FIELD_MIN} | - | - | 1 | ms |
| t _{Boot_RF_2} | RF 启动时间 ⁽⁵⁾ | TD_EVENT_UPDATE_EN=1b 或 UTC_EN=1b, 来自 H _{FIELD_MIN} | - | - | 5 | ms |
| t _{RF_OFF} | IC 复位所需的 RF 掉电持续时间 | - | 2 | - | - | ms |
| t ₁ | VICC 响应延迟 | - | 318.6 | 320.9 | 323.3 | μs |
| t ₂ | 收到 VICC 响应后的 VCD 新请求延迟 | - | 309 | 311.5 | 314 | μs |
| t ₃ | 未收到 VICC 响应后的 VCD 新请求延迟 | - | 323.3 | - | - | μs |

| 符号 | 说明 | 条件 ⁽¹⁾⁽²⁾ | 最小值 | 典型值 | 最大值 | 单位 |
|--------------------|----------------------------|-----------------------------|--------|-----|-------|------------|
| W_t | 写操作持续时间 ⁽⁶⁾ | 最大 32 位数据 | - | 4 | - | ms |
| C_{TUN} | 输入电容 ⁽⁴⁾⁽⁷⁾ | $f = 13.56 \text{ MHz}$ | 21.85 | 23 | 24.15 | pF |
| V_{BACK} | 最低 ISO15693 后向散射电压 | - | 10 | - | - | mV |
| $V_{MIN_1}^{(8)}$ | AC0 与 AC1 之间的最小输入电压幅度（峰到峰） | 盘点和读取操作 | - | 4.8 | - | V |
| $V_{MIN_2}^{(8)}$ | AC0 与 AC1 之间的最小输入电压幅度（峰到峰） | 写操作 | - | 6 | - | V |
| R_{closed} | 篡改回路闭合后的电阻 | TD0 和 TD1 已连接 | - | - | 50 | Ω |
| R_{open} | 篡改回路打开后的电阻 | TD0 和 TD1 未连接 | 1 | - | - | M Ω |
| t_{RET} | 保存期限 | $T_A \leq 55^\circ\text{C}$ | 60 | - | - | 年 |
| 循环 | 写循环可擦写次数 | $T_A \leq 85^\circ\text{C}$ | 100000 | - | - | 循环 |

1. $T_A = -40 - 85^\circ\text{C}$
2. 所有时间特性都是使用带如下特性的参考天线执行的：
 - ISO 天线 class1
 - 调谐频率 = 13.7 MHz
3. 在标准测试台上进行特性测试
4. 在 POR 电平下的对晶圆进行特性测试（仅限室温）
5. 载波生成到开始第一个请求的最短时间
6. 采用 1/4 编码的 VCD 请求，高数据率和单副载波下的 VICC 响应
7. 用于设计参考天线。最小值和最大值从工业测试仪的限制推导得出
8. (VAC0-VAC1)峰峰值特性测试

9 封装信息

为满足环境要求，意法半导体为这些器件提供了不同等级的 **ECOPACK** 封装，具体取决于它们的环保合规等级。ECOPACK 规范、等级定义和产品状态可在 www.st.com 网站获得。ECOPACK 是意法半导体的商标。

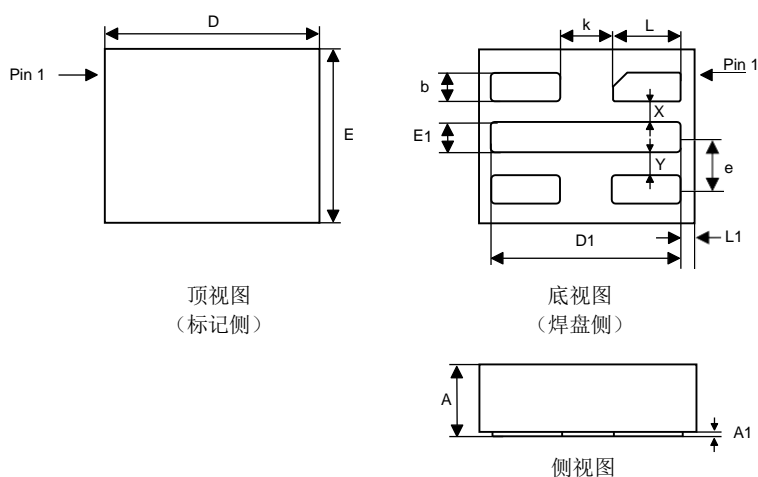
9.1 已切割植球晶圆

请联系意法半导体销售办事处获取说明文档。

9.2 UFDFPN5 (DFN5) 封装信息

UFDFPN5 是一种 5 引线、 1.7×1.4 mm、0.55 mm 厚的超薄紧密排列双扁平封装。

图 15. UFDFPN5 - 轮廓

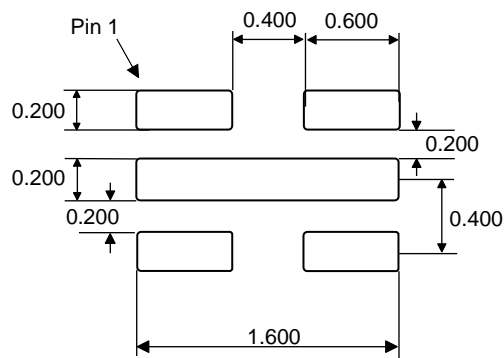


1. 最大封装翘曲为 0.05 mm。
2. 露铜并非系统性，根据横截面可能部分或全部出现。
3. 图纸未按比例绘制。
4. 在底侧，引脚 1 由特定的焊盘形状识别；在顶侧，引脚 1 根据标记的方向定义。读取标记时，引脚 1 在左上封装角的下面。

表 174. UFDFPN5 - 机械数据

| 符号 | 毫米 | | | 英寸 ⁽¹⁾ | | |
|---------------------|-------|-------|-------|-------------------|--------|--------|
| | 最小值 | 典型值 | 最大值 | 最小值 | 典型值 | 最大值 |
| A | 0.500 | 0.550 | 0.600 | 0.0197 | 0.0217 | 0.0236 |
| A1 | 0.000 | - | 0.050 | 0.0000 | - | 0.0020 |
| b 尺寸 ⁽²⁾ | 0.175 | 0.200 | 0.225 | 0.0069 | 0.0079 | 0.0089 |
| D | 1.600 | 1.700 | 1.800 | 0.0630 | 0.0669 | 0.0709 |
| D1 | 1.400 | 1.500 | 1.600 | 0.0551 | 0.0591 | 0.0630 |
| E | 1.300 | 1.400 | 1.500 | 0.0512 | 0.0551 | 0.0591 |
| E1 | 0.175 | 0.200 | 0.225 | 0.0069 | 0.0079 | 0.0089 |
| X | - | 0.200 | - | - | 0.0079 | - |
| Y | - | 0.200 | - | - | 0.0079 | - |
| e | - | 0.400 | - | - | 0.0157 | - |
| L | 0.500 | 0.550 | 0.600 | 0.0197 | 0.0217 | 0.0236 |
| L1 | - | 0.100 | - | - | 0.0039 | - |
| k | - | 0.400 | - | - | 0.0157 | - |

1. 英寸值由毫米值换算而来，四舍五入至 4 位小数。
2. b 用于镀层端子，测得其距端子顶部的距离在 0.15mm 和 0.30mm 之间。

图 16. UFDFPN5 - 建议封装图


提示 尺寸单位为毫米。

10 订购信息

表 175. 订货代码

| | | | | | | | |
|---|--------|-----|----|---|---|---|---|
| 示例: | ST25TV | 02K | C- | A | F | G | 3 |
| 设备类型 | | | | | | | |
| ST25TV = 基于 ISO 15693 和 NFC T5T 的 NFC/RFID 标签 | | | | | | | |
| 存储器容量 | | | | | | | |
| 512 = 512 位 | | | | | | | |
| 02K = 2560 位 | | | | | | | |
| 产品版本 | | | | | | | |
| C = 版本 C | | | | | | | |
| 接口 | | | | | | | |
| A = 无 | | | | | | | |
| T = 篡改检测 | | | | | | | |
| 特征 | | | | | | | |
| F = 增强型 NDEF | | | | | | | |
| 封装 | | | | | | | |
| F = 75 $\mu\text{m} \pm 10 \mu\text{m}$ 已切割植球晶圆 G | | | | | | | |
| = 120 $\mu\text{m} \pm 10 \mu\text{m}$ 已切割植球晶圆H = | | | | | | | |
| UFDFPN5 | | | | | | | |
| 电容 | | | | | | | |
| 3 = 23 pF | | | | | | | |

提示 标有“ES”或“E”的部件尚未通过认证，因此未获准用于生产。意法半导体对此类使用产生的任何后果概不负责。在任何情况下，意法半导体都不负责客户在生产中对这些工程样片的使用。在决定使用这些工程样例运行品质检测之前，必须联系意法半导体质量部门。

11 缩写语列表

表 176. 缩写语列表

| 缩写语 | 定义 |
|--------|--|
| AFI | 应用族标识符 |
| ANDEF | 增强型 NDEF |
| ASCII | 美国信息交换标准 |
| BSS | 块安全状态 |
| CC | 容量容器 |
| CMD | 指令 |
| CRC | 循环冗余校验 |
| DSFID | 数据存储格式标识符 |
| EEPROM | 电可擦除可编程只读存储器 |
| EOF | 帧结束 |
| FID | 特性标识符 |
| GDPR | 通用数据保护条例 |
| HZ | 高阻抗 |
| IC | 集成电路 |
| Id | 标识符 |
| NA | 不适用 |
| NC | 未连接 |
| NDEF | NFC 数据交换格式 |
| NFC | 近场通信 |
| PID | 参数标识符 |
| POR | 上电复位 |
| PWD | 密码 |
| RF | 射频 |
| RFID | RF 识别 |
| RFU | 保留供将来使用 |
| SOF | 帧起始 (Start of frame) |
| UFDFPN | 超薄紧密排列双扁平封装，无引脚 |
| UTC | 唯一输入码 |
| TD | 篡改检测 |
| UID | 唯一标识符 |
| VCD | 疏耦合设备 |
| VICC | 疏耦合集成电路卡 |
| X | 类型定义范围内的任何值（[0:1]适用于位，[0:F]适用于十六进制半字节） |

版本历史

表 177. 文档版本历史

| 日期 | 版本 | 变更 |
|------------------|----|---|
| 2020 年 12 月 15 日 | 1 | 初始版本。 |
| 2021 年 1 月 DD | 2 | 更新了： <ul style="list-style-type: none"> 第 5.2.2 节 唯一输入码描述 第 6.2.2 节 请求格式 第 6.2.4 节 响应格式 第 6.2.6 节 响应和错误码 第 10 节 订购信息 |

目录

| | | |
|----------|--------------|-----------|
| 1 | 说明 | 2 |
| 1.1 | 框图 | 2 |
| 1.2 | 封装连接 | 2 |
| 2 | 功能概述 | 4 |
| 2.1 | 框图 | 4 |
| 2.2 | 应用信息 | 4 |
| 3 | 电源管理 | 5 |
| 3.1 | 器件设置 | 5 |
| 3.2 | 器件复位 | 5 |
| 4 | 存储器管理 | 6 |
| 4.1 | 存储器组织结构 | 6 |
| 4.2 | 用户存储器 | 7 |
| 4.3 | 系统配置存储器 | 7 |
| 4.3.1 | 系统配置寄存器 | 8 |
| 4.3.2 | 系统寄存器 | 8 |
| 5 | 特性 | 10 |
| 5.1 | 数据保护 | 10 |
| 5.1.1 | 数据保护寄存器 | 10 |
| 5.1.2 | 密码管理 | 15 |
| 5.1.3 | 密码加密 | 18 |
| 5.1.4 | 用户存储器保护 | 19 |
| 5.1.5 | 系统配置存储器保护 | 19 |
| 5.2 | 唯一输入码 | 20 |
| 5.2.1 | 唯一输入码寄存器 | 20 |
| 5.2.2 | 唯一输入码描述 | 20 |
| 5.3 | 篡改检测 | 21 |
| 5.3.1 | 篡改检测寄存器 | 21 |
| 5.3.2 | 篡改检测说明 | 23 |
| 5.4 | 增强型 NDEF | 24 |

| | | |
|-------|------------------------------------|----|
| 5.4.1 | 增强型 NDEF 寄存器..... | 24 |
| 5.4.2 | 增强型 NDEF 描述 | 26 |
| 5.5 | 消费者隐私保护 | 28 |
| 5.5.1 | 增强型 NDEF messagePrivacy 寄存器示例..... | 28 |
| 5.5.2 | Kill 功能描述 | 29 |
| 5.5.3 | 不可追踪功能描述 | 29 |
| 5.6 | TruST25™数字签名 | 30 |
| 5.7 | AFI 保护 | 30 |
| 5.7.1 | AFI 保护寄存器 | 30 |
| 5.7.2 | AFI 保护说明 | 30 |
| 5.8 | Inventory Initiated | 31 |
| 5.9 | 设备识别寄存器..... | 31 |
| 6 | RF 操作..... | 34 |
| 6.1 | RF 通信 | 34 |
| 6.1.1 | 访问 ISO/IEC 15693 设备..... | 34 |
| 6.2 | 射频协议..... | 34 |
| 6.2.1 | 协议说明 | 34 |
| 6.2.2 | 请求格式 | 35 |
| 6.2.3 | 请求标志 | 35 |
| 6.2.4 | 响应格式 | 36 |
| 6.2.5 | 响应标志 | 37 |
| 6.2.6 | 响应和错误码 | 37 |
| 6.2.7 | 模式 | 38 |
| 6.2.8 | ISO15693 状态 | 38 |
| 6.2.9 | 自定义状态 | 39 |
| 6.3 | 时序定义..... | 41 |
| 6.4 | RF 指令 | 43 |
| 6.4.1 | Inventory | 43 |
| 6.4.2 | StayQuiet..... | 44 |
| 6.4.3 | ReadSingleBlock..... | 45 |
| 6.4.4 | WriteSingleBlock | 46 |
| 6.4.5 | LockBlock..... | 46 |

| | | |
|--------|--------------------------------------|----|
| 6.4.6 | ReadMultipleBlocks | 47 |
| 6.4.7 | Select | 48 |
| 6.4.8 | ResetToReady | 49 |
| 6.4.9 | WriteAFI | 49 |
| 6.4.10 | LockAFI | 50 |
| 6.4.11 | WriteDSFID | 51 |
| 6.4.12 | LockDSFID | 52 |
| 6.4.13 | GetSystemInfo | 52 |
| 6.4.14 | GetMultipleBlockSecurityStatus | 53 |
| 6.4.15 | ExtendedGetSystemInfo | 54 |
| 6.4.16 | ReadConfiguration | 55 |
| 6.4.17 | WriteConfiguration | 56 |
| 6.4.18 | WritePassword | 57 |
| 6.4.19 | PresentPassword | 59 |
| 6.4.20 | Kill | 60 |
| 6.4.21 | Initiate | 61 |
| 6.4.22 | InventoryInitiated | 61 |
| 6.4.23 | ToggleUntraceable | 62 |
| 6.4.24 | GetRandomNumber | 63 |
| 7 | 唯一标识符（UID） | 64 |
| 7.1 | 不可追踪的 UID | 64 |
| 8 | 设备参数 | 65 |
| 8.1 | 最大额定值 | 65 |
| 8.2 | RF 电气参数 | 65 |
| 9 | 封装信息 | 67 |
| 9.1 | 已切割植球晶圆 | 67 |
| 9.2 | UFDFPN5 封装信息 | 67 |
| 10 | 订购信息 | 69 |
| 11 | 缩略语列表 | 70 |
| | 版本历史 | 71 |
| | 目录 | 72 |



| | |
|-----------|----|
| 表一览 | 76 |
| 图一览 | 80 |

表一览

| | | |
|-------|--------------------------------|----|
| 表 1. | 信号名称 | 3 |
| 表 2. | 单区域模式中的用户存储器 | 7 |
| 表 3. | 双区域模式中的用户存储器 | 7 |
| 表 4. | 配置寄存器列表 | 8 |
| 表 5. | 系统寄存器列表 | 9 |
| 表 6. | LCK_CONFIG 访问 | 10 |
| 表 7. | LCK_CONFIG 内容 | 11 |
| 表 8. | LCK_BLOCK 访问 | 11 |
| 表 9. | LCK_BLOCK 内容 | 11 |
| 表 10. | RW_PROTECTION_A1 访问 | 12 |
| 表 11. | RW_PROTECTION_A1 内容 | 12 |
| 表 12. | END_A1 访问 | 12 |
| 表 13. | END_A1 内容 | 12 |
| 表 14. | RW_PROTECTION_A2 访问 | 13 |
| 表 15. | RW_PROTECTION_A2 内容 | 13 |
| 表 16. | PWD_CFG 访问 | 13 |
| 表 17. | PWD_CFG 内容 | 13 |
| 表 18. | PWD_A1 访问 | 14 |
| 表 19. | PWD_A1 内容 | 14 |
| 表 20. | PWD_A2 访问 | 14 |
| 表 21. | PWD_A2 内容 | 14 |
| 表 22. | PWD_UNTR 访问 | 14 |
| 表 23. | PWD_UNTR 内容 | 14 |
| 表 24. | RND_NUMBER 访问 | 15 |
| 表 25. | RND_NUMBER 内容 | 15 |
| 表 26. | 安全会话类型 | 15 |
| 表 27. | 密码寄存器列表 | 15 |
| 表 28. | RND_NUMBER_4B | 18 |
| 表 29. | RND_NUMBER_8B | 18 |
| 表 30. | 64 位 Password_data 值计算示例 | 18 |
| 表 31. | 块安全状态 | 19 |
| 表 32. | UTC_EN 访问 | 20 |
| 表 33. | UTC_EN 内容 | 20 |
| 表 34. | UTC 访问 | 20 |
| 表 35. | UTC 内容 | 20 |
| 表 36. | TD_EVENT_UPDATE_EN 访问 | 21 |
| 表 37. | TD_EVENT_UPDATE_EN 内容 | 21 |
| 表 38. | TD_SEAL_MSG 访问 | 21 |
| 表 39. | TD_SEAL_MSG 内容 | 21 |
| 表 40. | TD_UNSEAL_MSG 访问 | 21 |
| 表 41. | TD_UNSEAL_MSG 内容 | 22 |
| 表 42. | TD_RESEAL_MSG 访问 | 22 |
| 表 43. | TD_RESEAL_MSG 内容 | 22 |
| 表 44. | TD_SHORT_MSG 访问 | 22 |
| 表 45. | TD_SHORT_MSG 内容 | 22 |
| 表 46. | TD_OPEN_MSG 访问 | 22 |
| 表 47. | TD_OPEN_MSG 内容 | 22 |
| 表 48. | TD_STATUS 访问 | 23 |
| 表 49. | TD_STATUS 内容 | 23 |
| 表 50. | ANDEF_EN 访问 | 24 |
| 表 51. | ANDEF_EN 内容 | 24 |
| 表 52. | ANDEF_CFG 访问 | 24 |

| | | |
|--------|---|----|
| 表 53. | ANDEF_CFG 内容 | 24 |
| 表 54. | ANDEF_SEP 访问 | 25 |
| 表 55. | ANDEF_SEP 内容 | 25 |
| 表 56. | ANDEF_CUSTOM_LSB 访问 | 25 |
| 表 57. | ANDEF_CUSTOM_LSB 内容 | 25 |
| 表 58. | ANDEF_CUSTOM_MSB 访问 | 25 |
| 表 59. | ANDEF_CUSTOM_MSB 内容 | 25 |
| 表 60. | ANDEF_UID 访问 | 25 |
| 表 61. | ANDEF_UID 内容 | 26 |
| 表 62. | ST25TV02KC 上的 ANDEF 特性被禁用时的块数据读取 | 26 |
| 表 63. | ST25TV02KC 上的 ANDEF 特性被启用时的块数据读取 | 26 |
| 表 64. | 在 ANDEF_MEM 中串联的 ANDEF 字段 | 27 |
| 表 65. | KILL_CMD 访问 | 28 |
| 表 66. | KILL_CMD 内容 | 28 |
| 表 67. | UNTR_CMD 访问 | 28 |
| 表 68. | UNTR_CMD 内容 | 28 |
| 表 69. | PRIVACY 访问 | 28 |
| 表 70. | PRIVACY 内容 | 29 |
| 表 71. | AFI_PROT 访问 | 30 |
| 表 72. | AFI_PROT 内容 | 30 |
| 表 73. | LCK_DSFD 访问 | 31 |
| 表 74. | LCK_DSFD 内容 | 31 |
| 表 75. | LCK_AFI 访问 | 31 |
| 表 76. | LCK_AFI 内容 | 31 |
| 表 77. | DSFD 访问 | 32 |
| 表 78. | DSFD 内容 | 32 |
| 表 79. | AFI 访问 | 32 |
| 表 80. | AFI 内容 | 32 |
| 表 81. | IC_REF 访问 | 32 |
| 表 82. | IC_REF 内容 | 32 |
| 表 83. | REV 访问 | 33 |
| 表 84. | REV 内容 | 33 |
| 表 85. | UID 访问 | 33 |
| 表 86. | UID 内容 | 33 |
| 表 87. | 一般请求格式 | 35 |
| 表 88. | Request_flags LSB 的定义 | 36 |
| 表 89. | Inventory_flag 值为 0 时的 Request_flags MSB 定义 | 36 |
| 表 90. | Inventory_flag 值为 1 时的 Request_flags MSB 定义 | 36 |
| 表 91. | 一般响应格式 | 37 |
| 表 92. | Response_flags 的定义 | 37 |
| 表 93. | Error_flag 等于 1 时的一般响应格式 | 37 |
| 表 94. | 响应错误代码定义 | 37 |
| 表 95. | 取决于寻址模式的 Request_flags 值 | 39 |
| 表 96. | 取决于状态与寻址模式的器件响应 | 39 |
| 表 97. | 时序值 | 42 |
| 表 98. | 指令代码 | 43 |
| 表 99. | Inventory 请求格式 | 44 |
| 表 100. | Inventory 响应格式 | 44 |
| 表 101. | StayQuiet 请求格式 | 44 |
| 表 102. | ReadSingleBlock 请求格式 | 45 |
| 表 103. | Error_flag 等于 0 时的 ReadSingleBlock 响应格式 | 45 |
| 表 104. | Error_flag 等于 1 时的 ReadSingleBlock 错误代码 | 45 |
| 表 105. | WriteSingleBlock 请求格式 | 46 |
| 表 106. | Error_flag 等于 0 时的 WriteSingleBlock 响应格式 | 46 |

| | | |
|--------|---|----|
| 表 107. | Error_flag 等于 1 时的 WriteSingleBlock 错误代码..... | 46 |
| 表 108. | LockBlock 请求格式..... | 46 |
| 表 109. | Error_flag 等于 0 时的 LockBlock 响应格式..... | 47 |
| 表 110. | Error_flag 等于 1 时的 LockBlock 错误代码..... | 47 |
| 表 111. | ReadMultipleBlocks 请求格式..... | 47 |
| 表 112. | Error_flag 等于 0 时的 ReadMultipleBlocks 响应格式..... | 48 |
| 表 113. | Error_flag 等于 1 时的 ReadMultipleBlocks 错误代码..... | 48 |
| 表 114. | Select 请求格式..... | 48 |
| 表 115. | Error_flag 等于 0 时的选择响应格式..... | 48 |
| 表 116. | Error_flag 等于 1 时的 Select 错误代码..... | 49 |
| 表 117. | ResetToReady 请求格式..... | 49 |
| 表 118. | Error_flag 等于 0 时的 ResetToReady 响应格式..... | 49 |
| 表 119. | Error_flag 等于 1 时的 ResetToReady 错误代码..... | 49 |
| 表 120. | WriteAFI 请求格式..... | 50 |
| 表 121. | Error_flag 等于 0 时的 WriteAFI 响应格式..... | 50 |
| 表 122. | Error_flag 等于 1 时的 WriteAFI 错误代码..... | 50 |
| 表 123. | LockAFI 请求格式..... | 50 |
| 表 124. | Error_flag 等于 0 时的 LockAFI 响应格式..... | 50 |
| 表 125. | Error_flag 等于 1 时的 LockAFI 错误代码..... | 51 |
| 表 126. | WriteDSFID 请求格式..... | 51 |
| 表 127. | Error_flag 等于 0 时的 WriteDSFID 响应格式..... | 51 |
| 表 128. | Error_flag 等于 1 时的 WriteDSFID 错误代码..... | 51 |
| 表 129. | LockDSFID 请求格式..... | 52 |
| 表 130. | Error_flag 等于 0 时的 LockDSFID 响应格式..... | 52 |
| 表 131. | Error_flag 等于 1 时的 LockDSFID 错误代码..... | 52 |
| 表 132. | GetSystemInfo 请求格式..... | 52 |
| 表 133. | GetSystemInfo 响应..... | 53 |
| 表 134. | Error_flag 等于 1 时的 GetSystemInfo 错误代码..... | 53 |
| 表 135. | GetMultipleBlockSecurityStatus 请求格式..... | 53 |
| 表 136. | Error_flag 等于 0 时的 GetMultipleBlockSecurityStatus 响应格式..... | 53 |
| 表 137. | Error_flag 等于 1 时的 GetMultipleBlockSecurityStatus 错误代码..... | 54 |
| 表 138. | ExtendedGetSystemInfo 请求格式..... | 54 |
| 表 139. | Information_request_list 内容..... | 54 |
| 表 140. | Error_flag 等于 0 时的 ExtendedGetSystemInfo 响应格式..... | 54 |
| 表 141. | Information_flags 内容..... | 55 |
| 表 142. | Information_fields 内容..... | 55 |
| 表 143. | Error_flag 等于 1 时的 ExtendedGetSystemInfo 错误代码..... | 55 |
| 表 144. | ReadConfiguration 请求格式..... | 56 |
| 表 145. | Error_flag 等于 0 时的 ReadConfiguration 响应格式..... | 56 |
| 表 146. | Error_flag 等于 1 时的 ReadConfiguration 错误代码..... | 56 |
| 表 147. | WriteConfiguration 请求格式..... | 56 |
| 表 148. | Error_flag 等于 0 时的 WriteConfiguration 响应格式..... | 57 |
| 表 149. | Error_flag 等于 1 时的 WriteConfiguration 错误代码..... | 57 |
| 表 150. | WritePassword 请求格式..... | 57 |
| 表 151. | 密码寄存器列表..... | 58 |
| 表 152. | Error_flag 等于 0 时的 WritePassword 响应格式..... | 58 |
| 表 153. | Error_flag 等于 1 时的 WritePassword 错误代码..... | 58 |
| 表 154. | PresentPassword 请求格式..... | 59 |
| 表 155. | Error_flag 等于 0 时的 PresentPassword 响应格式..... | 59 |
| 表 156. | Error_flag 等于 1 时的 PresentPassword 错误代码..... | 59 |
| 表 157. | Kill 请求格式..... | 60 |
| 表 158. | Error_flag 等于 0 时的 Kill 响应格式..... | 60 |
| 表 159. | Error_flag 等于 1 时的 Kill 错误代码..... | 60 |
| 表 160. | Initiate 请求格式..... | 61 |

| | | |
|--------|--|----|
| 表 161. | Error_flag 等于 0 时的 Initiate 响应格式 | 61 |
| 表 162. | InventoryInitiated 请求格式 | 61 |
| 表 163. | Error_flag 等于 0 时的 InventoryInitiated 响应格式 | 61 |
| 表 164. | ToggleUntraceable 请求格式 | 62 |
| 表 165. | Error_flag 等于 0 时的 ToggleUntraceable 响应格式 | 62 |
| 表 166. | Error_flag 等于 1 时的 ToggleUntraceable 错误代码 | 63 |
| 表 167. | GetRandomNumber 请求格式 | 63 |
| 表 168. | Error_flag 等于 0 时的 GetRandomNumber 响应格式 | 63 |
| 表 169. | Error_flag 等于 1 时的 GetRandomNumber 错误代码 | 63 |
| 表 170. | UID 格式 | 64 |
| 表 171. | 不可追踪的 UID: UNTRACEABLE 状态中的 UID 值 | 64 |
| 表 172. | 绝对最大额定值 | 65 |
| 表 173. | RF 特性 | 65 |
| 表 174. | UFDFPN5 - 机械数据 | 68 |
| 表 175. | 订货代码 | 69 |
| 表 176. | 缩略语列表 | 70 |
| 表 177. | 文档版本历史 | 71 |

图一览

| | | |
|-------|---|----|
| 图 1. | ST25TVxxxC 框图 | 2 |
| 图 2. | UFDFPN5 封装连接 | 3 |
| 图 3. | 已切割植球晶圆的晶片连接 | 3 |
| 图 4. | RF 上电时序 | 5 |
| 图 5. | 存储器组织结构 | 6 |
| 图 6. | 安全会话管理 | 17 |
| 图 7. | ST25TV02KC-T 上的增强型 NDEF 消息示例 | 27 |
| 图 8. | ISO15693 协议时序 | 35 |
| 图 9. | ISO15693 状态转换图 | 39 |
| 图 10. | ST25TVxxxC 状态转换图 | 40 |
| 图 11. | VCD 与 ST25TVxxxC 之间的读之类帧交换 | 41 |
| 图 12. | Option_flag=0 时的 VCD 与 ST25TVxxxC 之间的写之类帧交换 | 42 |
| 图 13. | Option_flag=1 时 VCD 与 ST25TVxxxC 之间的写之类帧交换 | 42 |
| 图 14. | Stay Quiet 帧 | 45 |
| 图 15. | UFDFPN5 - 轮廓 | 67 |
| 图 16. | UFDFPN5 - 建议封装图 | 68 |

重要通知 - 请仔细阅读

意法半导体公司及其子公司（“意法半导体”）保留随时对意法半导体产品和/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  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。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。

© 2020 STMicroelectronics - 保留所有权利