AN14478

采用板卡控件扩展i.MX 9系统管理器功能

第1.0版—2024年11月25日 应用笔记

文档信息

信息	内容
关键词	AN14478, 系统管理器, i.MX 9, i.MX 94, i.MX 95
摘要	本应用笔记介绍了恩智浦6.6.36_2.1.0 BSP版本中提供的系统管理器（SM）及客户端的实现方法。

1 介绍

i.MX 9 系列中的一些处理器（例如 i.MX 95 和 i.MX 94）配备了专用的M33 内核，此内核专门运行系统管理器（System Manager，简称SM）。

SM是由恩智浦提供的固件，负责管理片上系统（SoC）中其他处理器的底层资源。它不仅提供了标准的Arm SCMI接口供客户端使用，还通过融入了一些恩智浦的专有功能进行了接口扩展。

SM以源代码的形式交付给客户，客户可以根据自身需求自由修改和重新编译其代码。然而，恩智浦建议客户对SM功能的任何扩展修改应仅限于客户电路板层，这种做法有效避免了直接修改SM内核代码可能带来的不必要的风险，同时实现了修改和扩展与内核代码的清晰隔离。

SM提供了一种灵活的机制，几乎可以重载和扩展其所有的入口点，并将这些入口点重定向至客户板层。尽管大多数SM公开的RPC（远程过程调用）都可以通过这种方式进行重载，但扩展SM功能的最佳方式仍然是添加专用的板卡控件。

本应用笔记详细介绍了恩智浦6.6.36_2.1.0 BSP版本中SM及其客户端的实现方法。

2 系统管理器控件

为了调整标准SCMI接口无法直接控制的设备或电路板设置，SM提供了对控件的访问权限。访问这些控件可以触发寄存器或变量的设置/获取操作，或执行更为复杂的任务。

每个控件均由一个32位的ID唯一标识。SM客户端可以使用恩智浦对SCMI协议的MISC扩展来操纵这些控件。

SM提供了一种便捷的扩展机制：客户只需修改他们的板层代码，即可添加自己的控件。这是扩展SM功能的首选方式。

3 管理控件的RPC调用

表1列出了SCMI API管理控件、它们的预期用途以及客户端调用它们所需的API权限。

表1. SCMI API管理控件

SCMI RPC	说明	用途	API权限
SCMI_MiscControlSet()	将控件数组设置为一组指定的值。	向SoC或外部外设寄存器写入数据。	SET（设置）
SCMI_MiscControlGet()	检索与控件相关的一组值。	从SoC或外部外设寄存器中读取数据。	GET（获取）
SCMI_MiscControlExtSet()	设置与控件相关的一组值，并将地址信息传递给SM。	将一系列值写入指定地址的SoC或外部外设寄存器。	SET（设置）
SCMI_MiscControlExtGet()	检索与控件相关的一组值，并将地址信息传递给SM。	从指定地址的SoC或外部外设寄存器读取一系列数值。	GET（获取）
SCMI_MiscControlAction()	执行与控件关联的任意操作。使用操作ID和参数列表。返回值列表。	此RPC可实现任何复杂的操作。	EXCLUSIVE（独占）

表1. SCMI API管理控件（续）

SCMI RPC	说明	用途	API权限
SCMI_MiscControlNotify()	注册代理，以便接收与控件相关的事件通知。	发生任意事件时，能够接收到其他逻辑机发出的事件通知。	NOTIFY（通知）
SCMI_MiscControlEvent()	广播与控件关联的事件，已注册的代理会收到通知。	发生任意事件时，可发送事件通知。	无

4 设备控件与板卡控件

对于每款支持i.MX的SoC，系统管理器（SM）都提供设备控件。这些控件主要用于控制标准SCMI协议未涵盖的SoC配置，例如全局设备配置寄存器的静态设置。设备控件是设备特定的，由SM代码提供。它们的ID可在`devices/<dev>/sm/dev_sm_control.h`中找到。与设备控件相关的RPC实现则位于`devices/<dev>/sm/dev_sm_control.c`文件中。

对于每块支持的板卡，SM同样会提供一套板卡控件。板卡控件用于公开SM管理的电路板相关组件。例如，SM可以使用板卡控件向SM客户端公开外部GPIO扩展器。板卡控件的实现位于`board/<board>/sm/brd_sm_control.[h|c]`文件中。

当存在板卡控件时，可以通过将相应的设备控件RPC重定向至板层，将它们暴露给客户端。有关详细的步骤，请参考《SM移植指南》：GitHub - nxp-imx/imx-sm：i.MX处理器的系统管理器固件及后续章节。

4.1 设备层重定向

SM的结构设计中，大多数访问客户端公开的设备资源的SCMI RPC调用均指向设备层中的函数。该层由恩智浦提供，仅针对特定设备，不得修改。

如果客户必须对现有SCMI协议实施特定扩展，SM会提供一种方法，可以将调用从设备层重定向到板层。通过这种方法，客户的特定实现就可以只驻留在板层中。

设备层重定向不仅限于SM控件。然而，将这一方法机制应用于SM控件是扩展SM功能的首选方法，同时它还将代码修改限制在板层中。

4.2 设备层重定向工作流程

RPC调用最终会调用逻辑机器管理（Logical Machine Management，LMM）层中的某个函数。这个函数要么重定向到设备层内的设备实现（即默认实现），要么重定向到板层内的自定义实现。

图1所示为LMM函数未被重定向到板层时的调用流程。

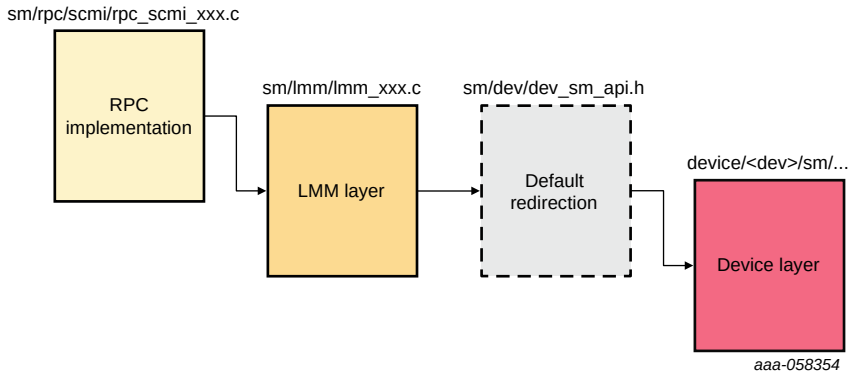

图1. LMM调用未被重定向的流程

图2所示为LMM函数重定向到板层时的调用流程。如果RPC调用的参数是此电路板特定的，例如访问板载传感器或板卡控件，则此板层实现必须执行特定操作。否则，此调用将回退到设备层实现。

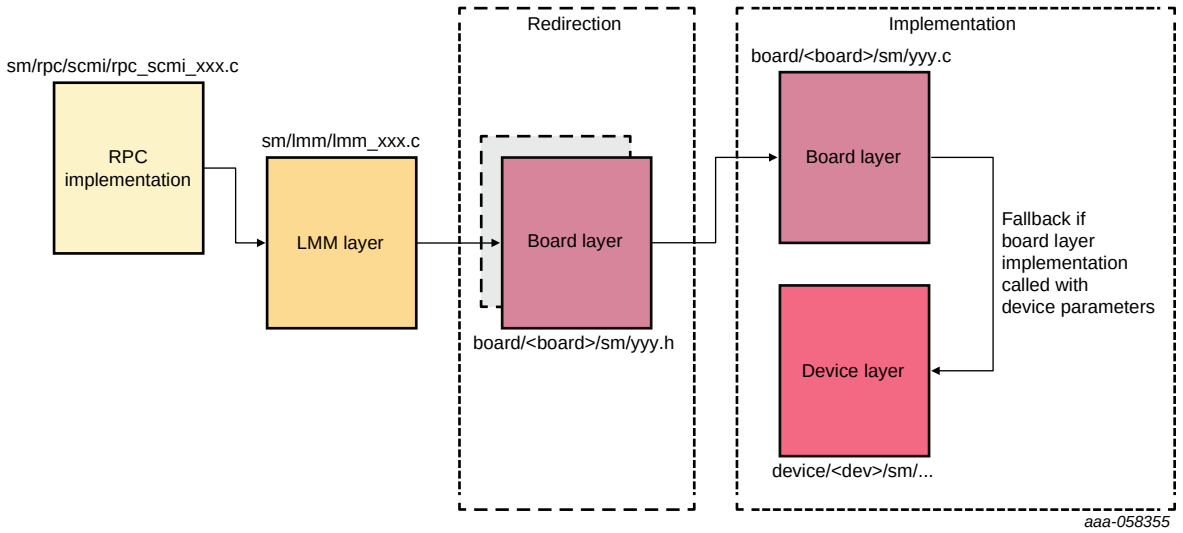

图2. LMM调用重定向到板层的流程

4.3 示例

图3以i.MX 95 SoC中的`SCMI_MiscControlSet()`为例，展示了重定向的过程。

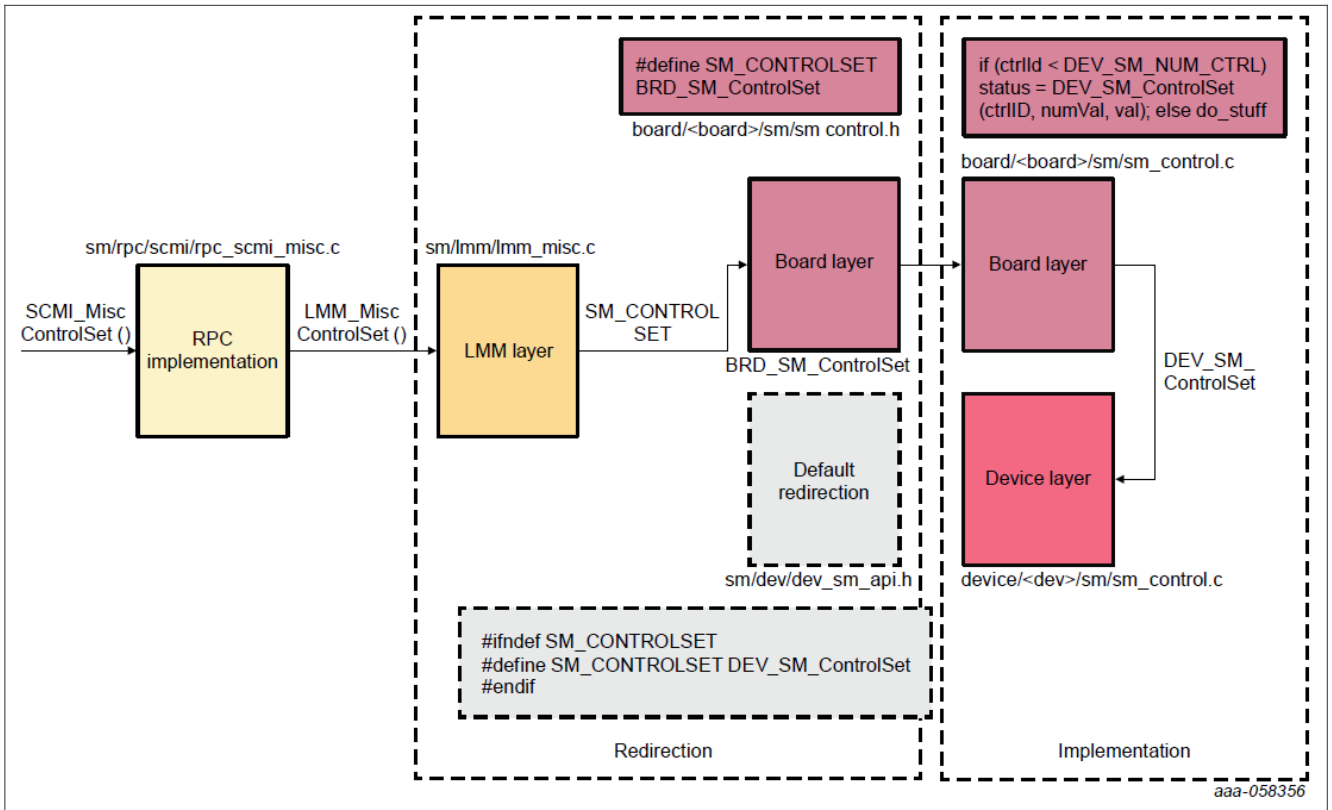

图3. SCMI_MiscControlSet()

- 从客户端调用`SCMI_MiscControlSet()`会执行`sm/lmm/lmm_misc.c`中定义的`LMM_MiscControlSet()`函数。
- `LMM_MiscControlSet()`调用`SM_CONTROLSET`。
- `SM_CONTROLSET`在`sm/dev/dev_sm_api.h`中定义如下：

```
#ifndef SM_CONTROLSET
/*! Redirector (device/board) to write a control */
#define SM_CONTROLSET           DEV_SM_ControlSet
#endif
```

- i.MX 95 EVK板层在`boards/mcimx95evk/sm/brd_sm_control.h`中重新定义了`SM_CONTROLSET`，以指向板层实现：

```
#define SM_CONTROLSET          BRD_SM_ControlSet
```

- 板卡控件ID及其编号也在`boards/mcimx95evk/sm/brd_sm_control.h`中进行了定义：

```
/*! Number of board controls */
#define BRD_SM_NUM_CTRL  7UL
/*! Total number of controls */
#define SM_NUM_CTRL  (DEV_SM_NUM_CTRL + BRD_SM_NUM_CTRL)
/*!
@name BRD_SM control domain indexes
*/
/** @{ */
#define BRD_SM_CTRL_SD3_WAKE     (DEV_SM_NUM_CTRL + 0U)
...
#define BRD_SM_CTRL_PCA2131      (DEV_SM_NUM_CTRL + 6U)
```

- 因此，RPC最终会调用BRD_SM_ControlSet。相应的实现位于boards/mcimx95evk/sm/brd_sm_control.c。如果此控件ID为设备控件ID，则返回设备实现；如果此控件ID为其他控件ID，它将提供板层实现。

```
int32_t BRD_SM_ControlSet(uint32_t ctrlId, uint32_t numVal, const uint32_t
 *val)
{
    int32_t status = SM_ERR_SUCCESS;
    /* Check to see if ctrlId is within bounds*/
    if (ctrlId < SM_NUM_CTRL)
    {
        /* Check if device or board */
        if (ctrlId < DEV_SM_NUM_CTRL)
        {
            status = DEV_SM_ControlSet(ctrlId, numVal, val);
        }
        else
        {
            /* Check the ctrlId and do what's needed */
        }
    }
    else
    {
        status = SM_ERR_NOT_FOUND;
    }
    /* Return status */
    return status;
}
```

- 为了让客户端能够操作新的板卡控件，必须获得访问权限。这些权限可在电路板配置文件configs/mx95evk.cfg中获得。

  注：*这些权限是按控件逐个授予的。*

```
#=================================================================#
# M7 EENV                                                         #
#============================== ==================================#
LM1    name="M7", rpc=scmi, boot=2, skip=1, did=4, safe=seenv
…
# API
…
BRD_SM_CTRL_PCA2131       ALL
…
```

5 板卡控件的实现

本节介绍了实现新板卡控件的过程，包括从控件ID的定义到在SM板层中进行的必要修改，逐步阐述了实现过程。

5.1 控件ID

为了避免设备控件和板卡控件之间的ID冲突，SM RPC层采用控件ID的第15位来区分这两种类型的控件。

- 如果控件 ID 的第 15 位未置位，则控件 ID 会直接传递到设备层（可能被重定向），无需任何修改。
- 如果控件 ID 的第 15 位已置位，则控件 ID 会以(input_control_id & ~0x8000) + DEV_SM_NUM_CTRL的形式传递到设备层（可能被重定向）。

采用第15位作为板卡控件区分标记的影响如下：

- 尽管CtrlID字段是一个32位无符号整数，但实际可用的控件ID范围限制为小于0xFFFF。这意味着32768个设备控件ID的范围从0x0000到0x7FFF，而板卡控件ID的范围从0x8000到0xFFFF。
- 在SM代码中，必须按照以下方式定义并使用板卡控件ID：

```
#define MY_ID (DEV_SM_NUM_CTRL + MY_INDEX)
```

- 在客户端代码中，向SCMI RPC传递给的板卡控件ID应为`MY_INDEX | MISC_CTRL_FLAG_BRD`。其中恩智浦客户端软件将`MISC_CTRL_FLAG_BRD`定义为`0x8000`。

5.2 实现概要

通过重定向MiscControlXXX()函数的设备实现，可以实现自定义板卡控件。

实现自定义板卡控件的步骤如下：

- 将请求 `SCMI_MiscControlxxx()` API 从设备层重定向到板层。在板层代码中重新定义 `SM_CONTROLXXX` 符号。
- 在板卡控件层中添加新的控件 ID，定义 `BRD_SM_MY_ID (DEV_SM_NUM_CTRL + BRD_SM_INDEX)`，并相应递增 `BRD_SM_NUM_CTRL` 的值。
- 授予客户端访问新控件的权限。在受影响的代理中为 `BDR_SM_MY_ID` 添加所需的 API 权限。
- 提供重定向函数的实现，或扩展现有实现。通过验证 `BRD_SM_MY_ID` 来检测目标是否为新控件。

5.2.1 客户端实现

在调用SCMI访问函数时，需将`BRD_SM_INDEX | MISC_CTRL_FLAG_BRD`作为CtrlID传递。

下面是客户端调用MiscControlAction控件操作的示例，该操作传递action=0，且无输入参数。

```
err = SCMI_MiscControlAction(SCMI_A2P, BRD_SM_INDEX | SCMI_MISC_CTRL_FLAG_BRD,
 0, 0, NULL, &numRtn,  &rtn);
```

5.3 注意事项

板卡控件和设备层重定向可以让任何客户端在SM环境中运行任意代码。在编写此类代码前，必须注意以下几点：

1. 限制板层代码的处理时间。SM是单线程模式。如果正在为一个SCMI调用占用过多时间，其他并发SCMI调用必须等待直到它返回。这个过程可能导致其他客户端长时间等待。在生产环境中，板层调用的处理时间建议控制在几百微秒或更短。例如，访问设备寄存器或执行单次快速的I2C读取可能没问题，但从慢速设备中转存大量I2C寄存器则不可取。为了评估对它们对SM响应速度的影响，请确保对板卡控件的实现进行性能分析。
2. 板层代码中的错误可能会导致SM挂起或崩溃，进而引发整个平台的复位。因此，板卡控件操作代码必须经过仔细审核。板卡控件函数（尤其是ControlAction）的输入必须经过严格清理和验证，以防止有漏洞或恶意客户端以不必要的方式对SM及整个平台造成不良影响。

5.4 板卡控件使用示例

1. 读取PF09 PMIC寄存器：
   - 创建BRD_SM_CTRL_PMIC_ACCESS控件ID。
   - 扩展SCMI_MiscControlExtGet(uint32_t ctrlId, uint32_t addr, uint32_t numRtn, uint32_t *rtn)的电路板实现，当采用BRD_SM_CTRL_PMIC_ACCESS ID调用时，调用BRD_SM_PmicRead(BOARD_PF09_DEV_ADDR, addr, rtn)。
     numRtn必须为1，且rtn必须是有效指针。
   - 任何客户端都可以按如下方式调用该函数。

```
SCMI_MiscControlExtGet(SCMI_A2P, BRD_SM_CTRL_PMIC_ACCESS |
 SCMI_MISC_CTRL_FLAG_BRD, pmic_reg_address, 1, &rtn);
```

2. 写入PF09 PMIC寄存器：
   - 使用与读取相同的控件ID。
   - 扩展SCMI_MiscControlExtSet()的电路板实现，使用BRD_SM_CTRL_PMIC_ACCESS ID调用时，向PMIC寄存器写入数据。

6 设备控件和板卡控件的客户端支持

i.MX MCUXpresso SDK v16_00 for i.MX 95支持完整的SCMI_MiscControl API集，但不包括SCMI_MiscControlExtSet()和SCMI_MiscControlExtGet()。

NXP Linux内核6.6.36_2.1.0仅支持SCMI_MiscControlSet、SCMI_MiscControlGet和SCMI_MiscControlNotify API。

7 控件特性

本节介绍了某些SM控件特有的参数和使用方法。

任何代理都可以调用SCMI_MiscCtrlNotify()来请求对controlID 的通知。然后，其他代理可以使用SCMI_MiscCtrlEvent()来广播与该controlID相关的通知。

SCMI_MiscCtrlNotify()和SCMI_MiscCtrlEvent()将接收控件ID和一个32位标志作为参数。SCMI_MiscCtrlEvent()发送的通知会传递给所有已为相关控件ID注册了通知的代理，这些代理至少有一个标志位与SCMI_MiscCtrlNotify()中设置的标志相匹配。

与SCMI_MiscCtrlNotify()相关的设备层调用是DEV_SM_ControlFlagSet()。此调用支持常规方式的重定向机制。i.MX 95的板端口移植使用该机制来通知代理GPIO扩展器的传入中断（具体实现可参见BRD_SM_ControlHandler()函数）。

8 参考资料

本文所引用的参考资料如下：

- Arm系统控制和管理接口平台设计文档v.3.2：https://developer.arm.com/documentation/den0056
- 系统管理器GitHub存储库：GitHub - nxp-imx/imx-sm: System Manager firmware for i.MX processors

9 缩略语

表2列出了本文中使用的缩略语。

表2. 缩略语

术语	说明
LMM	逻辑机器管理
RPC	远程过程调用
SCMI	系统控制和管理接口
SM	系统管理器
SoC	片上系统

10 关于本文中源代码的说明

本文中所示的示例代码具有以下版权和BSD-3-Clause许可：

2024年恩智浦版权所有；在满足以下条件的情况下，可以源代码和二进制文件的形式重新分发和使用本源代码（无论是否经过修改）：

1. 重新分发源代码必须保留上述版权声明、这些条件和以下免责声明。
2. 以二进制文件形式重新分发时，必须在文档和/或随分发提供的其他材料中复制上述版权声明、这些条件和以下免责声明。
3. 未经事先书面许可，不得使用版权所有者的姓名或参与者的姓名为本软件的衍生产品进行背书或推广。

本软件由版权所有者和参与者“按原样”提供，不承担任何明示或暗示的担保责任，包括但不限于对适销性和特定用途适用性的暗示保证。在任何情况下，无论因何种原因或根据何种法律条例，版权所有者或参与者均不对因使用本软件而导致的任何直接、间接、偶然、特殊、惩戒性或后果性损害（包括但不限于采购替代商品或服务；使用损失、数据损失或利润损失或业务中断）承担责任，无论是因合同、严格责任还是侵权行为（包括疏忽或其他原因）造成的，即使事先被告知有此类损害的可能性也不例外。

11 修订历史

表3汇总了本文档的修订情况。

表3. 修订历史

文档编号	发布日期	说明
AN14478 v.1.0	2024年11月25日	首次公开发布

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com.cn/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

目录

1 介绍 2
2 系统管理器控件 2
3 管理控件的RPC调用 2
4 设备控件与板卡控件 3
4.1 设备层重定向 3
4.2 设备层重定向工作流程 3
4.3 示例 4
5 板卡控件的实现 6
5.1 控件ID 6
5.2 实现概要 7
5.2.1 客户端实现 7
5.3 注意事项 7
5.4 板卡控件使用示例 8
6 设备控件和板卡控件的客户端支持 8
7 控件特性 8
8 参考资料 8
9 缩略语 8
10 关于本文中源代码的说明 9
11 修订历史 9
法律声明 10