

RG50xQ&RM5xxQ 系列

USB 虚拟光盘配置指导

5G 模块系列

版本：1.0

日期：2021-05-20

状态：受控文件



上海移远通信技术股份有限公司始终以为客户提供最及时、最全面的服务为宗旨。如需任何帮助，请随时联系我司上海总部，联系方式如下：

上海移远通信技术股份有限公司
上海市闵行区田林路 1016 号科技绿洲 3 期（B 区）5 号楼 邮编：200233
电话：+86 21 51086236 邮箱：info@quectel.com

或联系我司当地办事处，详情请登录：<http://www.quectel.com/cn/support/sales.htm>。

如需技术支持或反馈我司技术文档中的问题，可随时登陆如下网址：
<http://www.quectel.com/cn/support/technical.htm> 或发送邮件至：support@quectel.com。

前言

上海移远通信技术股份有限公司提供该文档内容用以支持其客户的产品设计。客户须按照文档中提供的规范、参数来设计其产品。因未能遵守有关操作或设计规范而造成的损害，上海移远通信技术股份有限公司不承担任何责任。在未声明前，上海移远通信技术股份有限公司有权对该文档进行更新。

免责声明

上海移远通信技术股份有限公司尽力确保开发中功能的完整性、准确性、及时性或效用，但不排除上述功能错误或遗漏的可能。除非其他有效协议另有规定，否则上海移远通信技术股份有限公司对开发中功能的使用不做任何暗示或明示的保证。在适用法律允许的最大范围内，上海移远通信技术股份有限公司不对任何因使用开发中功能而遭受的损失或损害承担责任，无论此类损失或损害是否可以预见。

保密义务

除非上海移远通信技术股份有限公司特别授权，否则我司所提供文档和信息的接收方须对接收的文档和信息保密，不得将其用于除本项目的实施与开展以外的任何其他目的。未经上海移远通信技术股份有限公司书面同意，不得获取、使用或向第三方泄露我司所提供的文档和信息。对于任何违反保密义务、未经授权使用或以其他非法形式恶意使用所述文档和信息的违法侵权行为，上海移远通信技术股份有限公司有权追究法律责任。

版权申明

本文档版权属于上海移远通信技术股份有限公司，任何人未经我司允许而复制转载该文档将承担法律责任。

版权所有 ©上海移远通信技术股份有限公司 2021，保留一切权利。

Copyright © Quectel Wireless Solutions Co., Ltd. 2021.

文档历史

修订记录

版本	日期	作者	变更表述
-	2020-12-10	Vern LIN	文档创建
1.0	2021-05-20	Vern LIN	受控版本

目录

文档历史	2
目录	3
表格索引	4
图片索引	5
1 引言	6
1.1. 适用模块	6
2 从设备 Gadget 概述.....	7
2.1. 主机 Host 与从设备 Gadget 工作方式	7
2.2. USB Gadget 软硬件构成	8
2.3. Gadget 设备驱动配置方法	9
3 USB Gadget 虚拟光盘配置	10
3.1. 配置前提	10
3.2. 脚本配置步骤.....	10
3.3. 参考脚本应用.....	12
3.4. 注意事项	13
4 USB Gadget 虚拟光盘识别	14
4.1. 开启 USB Gadget 虚拟光盘	14
4.2. 关闭 USB Gadget 虚拟光盘	15
5 USB Gadget 虚拟光盘系统兼容性验证	16
6 附录：术语缩写	17

表格索引

表 1: 适用模块	6
表 2: USB Gadget 虚拟光盘系统兼容性验证	16
表 3: 术语缩写	17

图片索引

图 1：主机 Host 与从设备 Gadget 工作方式	7
图 2：USB 堆栈软件架构	8
图 3：开启 Linux File-CD Gadget USB Device.....	14
图 4：关闭 Linux File-CD Gadget USB Device.....	15

1 引言

本文档主要介绍了如何通过 Linux *configs* 配置移远通信 RG50xQ 和 RM5xxQ 系列模块的 USB Gadget 虚拟光盘功能，以及虚拟光盘的识别方法和系统兼容性验证结果。

1.1. 适用模块

表 1：适用模块

模块系列	模块
RG50xQ	RG500Q 系列
	RG501Q-EU
	RG502Q-EA
RM5xxQ	RM500Q 系列
	RM502Q 系列
	RM505Q-AE
	RM510Q-GL

2 从设备 Gadget 概述

2.1. 主机 Host 与从设备 Gadget 工作方式

Linux 支持连接各种 USB 从设备，同时也支持作为设备插入到其他主机中，例如，Android 手机连接到电脑会被识别为 U 盘类的设备。从硬件角度而言，USB 从设备直接对应 USB 设备端。

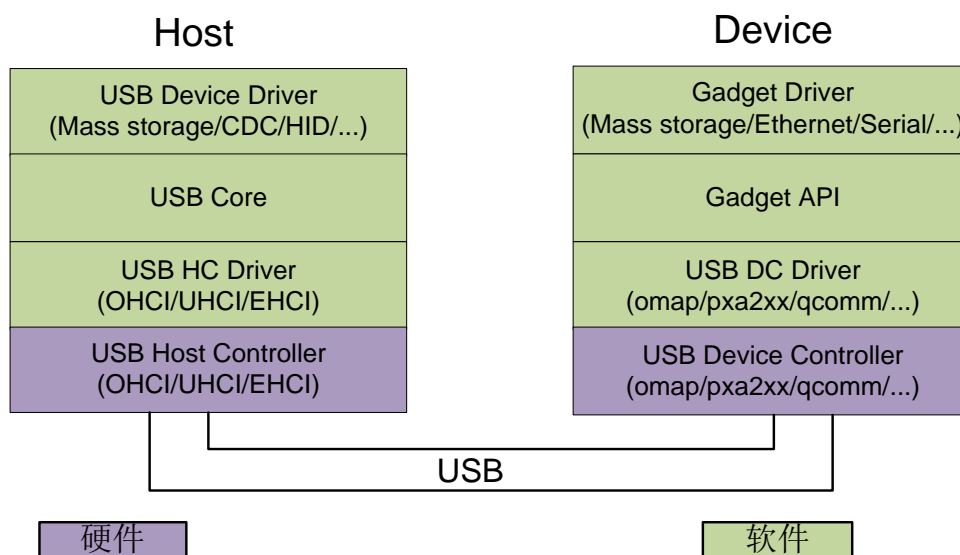


图 1：主机 Host 与从设备 Gadget 工作方式

Linux 支持 USB 主机驱动，同时也支持从设备驱动，为避免与作为主机时支持的“USB 设备驱动（USB Device Driver）”一词混淆，Linux 给 USB 从设备的实现取名为“Gadget”。Linux 内核源码路径 `drivers/usb/gadget` 中包含了内核支持的不同类型 USB 设备控制器(UDC)驱动、Gadget 框架和各种 Gadget 功能的实现。

作为 PC 的外设，模块通过 USB 数据线连接到 PC 后，模块处于 USB Device 模式，即 USB Gadget 工作模式。USB Gadget 功能，从主机端来看，它是具有一系列配置（configurations）的集合，每个配置含有多个接口（interfaces）；从 USB 设备端来看，接口（interface）即功能（function），每个接口对应一个功能，如串口、ECM 和大容量存储等。

2.2. USB Gadget 架构

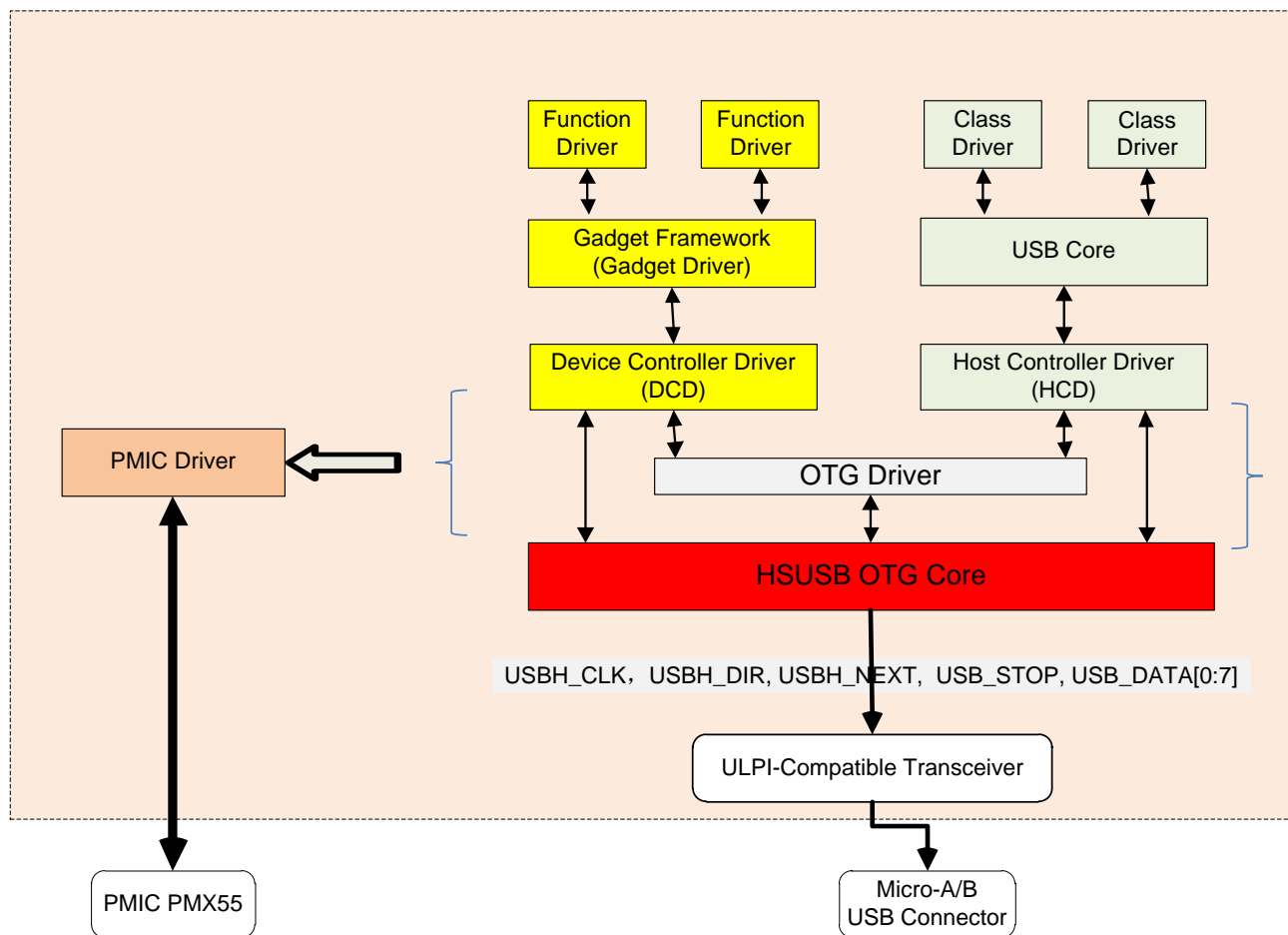


图 2: USB 堆栈

USB Gadget 从设备主要是通过 Gadget API 实现，其软件架构分为以下三个层次：

- Function Driver, 对应 USB 功能，即按照 USB Class 协议 USB Gadget 从设备需要实现的功能（如 HID、MSC、ECM 等）。
- Gadget Framework, 即 Linux 提供的 USB Gadget 驱动框架。
- DCD, 即 USB 设备控制器驱动，用来实现控制器相关驱动。

2.3. Gadget 设备驱动配置方法

USB Gadget 设备驱动有以下三种配置方法：

- 注册 USB composite 驱动
- 通过 Android FunctionFS 用户态驱动进行配置
- 通过 Linux *configfs* 进行配置

本文档所介绍的 USB Gadget 虚拟光盘配置是基于 Linux *configfs* 进行的。

configfs 是一个基于内存的虚拟文件系统，它提供的功能与 *sysfs* 相反。*sysfs* 是为了导出内核对象给用户空间查看；而 *configfs* 则是为了从用户空间配置内核对象，且不用重编内核与修改内核代码。用户态利用 *configfs* 文件系统创建 USB 设备对应的配置、接口及端点等，并将其绑定到 UDC 驱动，从而实现某种 USB Gadget 设备。

3 USB Gadget 虚拟光盘配置

本章节主要介绍如何通过 Linux *configs* 将 USB 大容量存储功能配置为虚拟光盘功能。USB Gadget 中大容量存储功能的相关参数及配置，请参考内核源码 *Documentation/USB/massstorage.txt*。

3.1. 配置前提

要配置虚拟光盘功能，Linux 内核需支持 USB Gadget 设备的如下配置：

```
CONFIG_USB_GADGET=y
CONFIG_USB_LIBCOMPOSITE=y
CONFIG_USB_F_MASS_STORAGE=y
CONFIG_USB_F_FS=y
CONFIG_USB_CONFIGFS=y
```

上述内核配置在 RG50xQ 和 RM5xxQ 系列模块内核中均已支持，并已作为内嵌功能编译进模块内核中。

备注

若出现虚拟光盘功能不能使用的情况，需先确认上述内核配置是否为模块默认的打开状态。

3.2. 脚本配置步骤

脚本配置步骤如下：

1. 检测 *configs* 是否挂载；

```
~ # mount | grep configs
configs on /sys/kernel/config type configs (rw,relatime)
```

上述执行结果显示 *configs* 挂载点在 */sys/kernel/config* 目录上。

2. 切换到 `usb_gadget` 实例目录：

```
~ # cd /sys/kernel/config/usb_gadget/g1
```

3. 检查 `mass_storage.0` 是否存在：

```
~ # ls -la ../g1/functions/mass_storage.0
```

若不存在 `mass_storage.0`，则使用如下命令建立一个 `mass_storage.0` 对象：

```
mkdir /sys/kernel/config/usb_gadget/g1/functions/mass_storage.0
```

4. 设置虚拟光盘功能开关：

- 1) 设置 “/” 根分区为 `rw` 属性

```
~ # mount -o remount, rw /
```

- 2) 创建配置文件

创建一个配置文件作为配置开关，并存放在 `/data/usb` 目录。配置开关设置：1 表示开启；0 表示关闭（如下代码示例中，配置开关默认为开启状态）。

```
~ # echo 1 > /data/usb/quec_usb_cdrom
```

5. 修改 `/sbin/usb/compositions/90DB` shell 脚本以实现配置：

- 1) 读取光盘功能开关配置

```
if [ -f /data/usb/quec_usb_cdrom ]
then
    QUCETEL_USB_CDROM=`cat /data/usb/quec_usb_cdrom`
else
    echo "[!xf][QUCETEL_USB_CDROM] QUCETEL_USB_CDROM NOT exists!" > /dev/kmsg
fi
```

- 2) 配置光盘及对应挂载文件

```
if [ "$QUCETEL_USB_CDROM" == "1" ]
then
    if [ ! -d /sys/kernel/config/usb_gadget/g1/functions/mass_storage.0 ]
    then
        mkdir /sys/kernel/config/usb_gadget/g1/functions/mass_storage.0
```

```

fi

echo "" > /sys/kernel/config/usb_gadget/g1/functions/mass_storage.0/lun.0/file
echo 1 > /sys/kernel/config/usb_gadget/g1/functions/mass_storage.0/lun.0/cdrom
echo "/sbin/usb/compositions/cdrom.iso" >
/sys/kernel/config/usb_gadget/g1/functions/mass_storage.0/lun.0/file
#add end by vern
fi

```

若不存在 *cdrom.iso*（格式：ISO 9660）镜像，可使用如下命令将镜像文件存放到 Ubuntu 对应目录中：

```
~ # mkisofs -r -o cdrom.iso /your-cdrom-dir
```

3) 关联 *mass_storage.0* 功能和 *configs* 配置

```

if [ "$QUCETEL_USB_CDROM" == "1" ];then
    ln -s functions/mass_storage.0 configs/c.1/f$func_index
    func_index=$((func_index+1))
fi

```

6. 执行 **sync** 命令保存 *90DB* 和 *quec_usb_cdrom* 文件。

3.3. 参考脚本应用

用户也可以直接参考移远通信提供的参考脚本 *90DB* 及配置文件 *quec_usb_cdrom* 来配置 USB 虚拟光盘功能，单击下方图标以获取相关文件：



usb_cdrom.rar

- *90DB* 为 USB 复合功能（USB composition）接口配置脚本，系统路径为 */sbin/usb/compositions*；
- *quec_usb_cdrom* 为 USB 虚拟光盘功能配置开关文件，存放在 */data/usb* 目录即可。

90DB 脚本读取 *quec_usb_cdrom* 中的配置开关 1 或 0 来打开或关闭 USB 虚拟光盘功能。

参考脚本的具体使用方式如下：

1. 将上述附件解压，即可得到参考脚本 *90DB* 及配置文件 *quec_usb_cdrom*;
2. 参考 *90DB* 脚本文件，搜索注释关键词“vern”，即可找到如下相关配置语句：

```
#added by vern for cdrom feature switch check
if [ -f /data/usb/quec_usb_cdrom ]
then
    QUCETEL_USB_CDROM=`cat /data/usb/quec_usb_cdrom`
else
    echo "[!xf][QUCETEL_USB_CDROM] QUCETEL_USB_CDROM NOT exists!" > /dev/kmsg
fi
#added end by vern
```

```
if [ "$QUCETEL_USB_CDROM" == "1" ]
then
if [ ! -d /sys/kernel/config/usb_gadget/g1/functions/mass_storage.0 ]
then
mkdir /sys/kernel/config/usb_gadget/g1/functions/mass_storage.0
fi
echo "" > /sys/kernel/config/usb_gadget/g1/functions/mass_storage.0/lun.0/file
echo 1 > /sys/kernel/config/usb_gadget/g1/functions/mass_storage.0/lun.0/cdrom
echo "/sbin/usb/compositions/cdrom.iso"
> /sys/kernel/config/usb_gadget/g1/functions/mass_storage.0/lun.0/file
fi
```

```
if [ "$QUCETEL_USB_CDROM" == "1" ]
then
ln -s functions/mass_storage.0 configs/c.1/f$func_index
func_index=$((func_index+1))
fi
```

3. 通过 USB 串口实现 SSH 工具（如 SecureCRT 或 MobaXterm）与模块之间的通信连接；
4. 设置“/”根分区为 rw 属性；
5. 使用 vi 工具添加上述配置语句到 */sbin/usb/compositions/90DB* 文件；
6. 执行 **echo** 命令创建配置文件 *quec_usb_cdrom*;
7. 执行 **sync** 命令保存 *90DB* 和 *quec_usb_cdrom* 文件。

3.4. 注意事项

USB 设备端虚拟光盘功能仅能与 RmNet 网卡功能（配置脚本为 *90DB*）共存，但不能与 USB ECM 或 RNDIS 网卡功能（配置脚本为 *9059*）共存，因此，请勿在 *9059* 脚本中配置 USB 虚拟光盘功能。

4 USB Gadget 虚拟光盘识别

本章节主要介绍如何在 Windows PC 上识别 USB Gadget 虚拟光盘。

4.1. 开启 USB Gadget 虚拟光盘

使用如下命令开启 USB Gadget 虚拟光盘功能：

```
~ # echo 1 > /data/usb/quec_usb_cdrom  
~ # sync
```

重启模块，然后将模块通过 USB 数据线连接到 Windows PC，此时查看设备管理器或文件浏览器应存在 Linux File-CD Gadget USB Device，如下图所示：



图 3：开启 Linux File-CD Gadget USB Device

4.2. 关闭 USB Gadget 虚拟光盘

使用如下命令关闭 USB Gadget 光盘功能:

```
~# echo 0 > /data/usb/quec_usb_cdrom
~# sync
```

重启模块，确保模块已通过 USB 数据线连接到 Windows PC，此时查看设备管理器或文件浏览器应不存在 Linux File-CD Gadget USB Device，如下图所示：

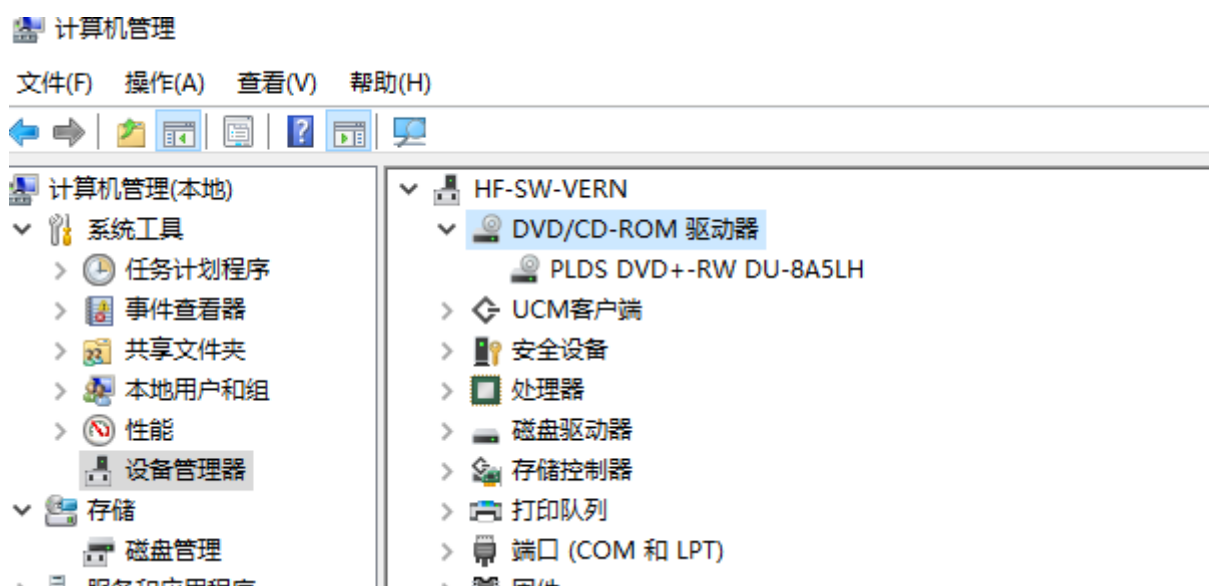


图 4：关闭 Linux File-CD Gadget USB Device

5 USB Gadget 虚拟光盘系统兼容性验证

移远通信验证了 RG50xQ 和 RM5xxQ 系列模块 USB 虚拟光盘功能的系统兼容性，详细验证结果如下：

表 2：USB Gadget 虚拟光盘系统兼容性验证

操作系统	验证版本	验证结果
Windows	<ul style="list-style-type: none"> Windows 10 Windows 7 	1. 能够正常识别光盘； 2. 光盘功能启用时，其他 USB 设备功能正常。 <ul style="list-style-type: none"> 通过 QCOM 工具 AT 端口能够正常发送 AT 命令 通过 QFlash 工具 DM 端口能够正常下载镜像文件 ADB 功能能够正常开关；通过 ADB 能够正常传送（push/pull）文件
	1. Linux: Ubuntu 14.04	
Linux	2. VMware 虚拟机: Ubuntu 14.04 （基于 Windows 10） （必须先卸载模块的 USB 驱动，虚拟机才能识别光盘）	能够正常识别光盘
Mac	Mac OS 10.15.2	不支持

6 附录：术语缩写

表 3：术语缩写

术语	英文全称	中文全称
ADB	Android Debug Bridge	Android 调试桥
API	Application Programming Interface	应用程序接口
CD	Compact Disc	光盘
CDC	Communications Device Class	通信设备类
CDROM	Compact Disc Read-Only Memory	只读光盘
DC	Device Controller	设备控制器
DCD	Device Controller Driver	设备控制器驱动
DM	Diagnostic Monitor	诊断监控
ECM	Ethernet Control Module	以太网控制模块
EHCI	Enhanced Host Controller Interface	增强型主机控制器接口
FS	File System	文件系统
HC	Host Controller	主机控制器
HCD	Host Controller Driver	主机控制器驱动
HID	Human Interface Device	人机接口设备
HSUSB	Hi-Speed USB	高速 USB
ISO	International Organization for Standardization	国际标准化组织
MSC	Mass Storage Class	大容量存储设备类
OHCI	Open Host Controller Interface	开放式主机控制器接口
OS	Operation System	操作系统

OTG	On-The-Go	USB OTG 补充标准
PC	Personal Computer	个人电脑
PMIC	Power Management Integrated Circuit	电源管理芯片
RNDIS	Remote Network Driver Interface Specification	远程网络驱动程序接口规范
SSH	Secure Shell	安全 Shell 协议
UDC	USB Device Controller	USB 设备控制器
UHCI	Universal Host Controller Interface	通用主机控制器接口
ULPI	UTMI+ Low Pin Interface	UTMI+低引脚数接口
USB	Universal Serial Bus	通用串行总线
UTMI	USB 2.0 Transceiver Macrocell Interface	USB 2.0 传输宏接口