



XR806 Mesh 控制台命令 使用指南

版本号：1.0

发布时间：2021-03-25

版本历史

版本	日期	责任人	版本描述
1.0	2021-03-25	AWA 1451	创建文档。

目录

版本历史.....	i
目录.....	ii
表格目录.....	vi
1 前言.....	1
1.1 文档简介.....	1
1.2 目标读者.....	1
1.3 适用范围.....	1
1.4 文档约定.....	1
1.4.1 标志说明.....	1
2 概述.....	2
2.1 背景说明.....	2
2.2 规格特性.....	2
2.3 文件位置.....	2
3 应用说明.....	3
3.1 应用简述.....	3
3.2 配置说明.....	3
3.3 格式说明.....	3
3.4 命令说明.....	4
3.4.1 通用配置命令.....	4
3.4.1.1 init（mesh 初始化）.....	4
3.4.1.2 reset（重置 mesh 节点）.....	4
3.4.1.3 lpm（启用/关闭节点低功耗功能）.....	5
3.4.1.4 poll（从朋友节点获取数据）.....	5
3.4.1.5 ident（启用节点身份认证广播）.....	5
3.4.1.6 dst（设置操作的节点对象）.....	6
3.4.1.7 netidx（设置 mesh Netkey 索引）.....	6
3.4.1.8 appidx（设置 mesh Appkey 索引）.....	6
3.4.1.9 net-send（网络内发送数据）.....	6
3.4.1.10 iv-update（切换 IV 更新状态）.....	7
3.4.1.11 iv-update-test（使能/关闭 IV 更新测试）.....	7
3.4.1.12 rpl-clear（清空重传保护列表）.....	7

3.4.1.13 pb-gatt (PB-GATT 广播)	8
3.4.1.14 pb-adv (PB-ADV 广播)	8
3.4.1.15 uuid (设置 mesh 设备 UUID)	8
3.4.1.16 input-num (输入式 OOB 认证数字输入)	9
3.4.1.17 input-str (输入式 OOB 认证字符串输入)	9
3.4.1.18 static-oob (静态 OOB 值设置)	9
3.4.1.19 prov-method (配网认证方式配置)	9
3.4.1.20 provision (配网者将自身配入到网络)	10
3.4.1.21 timeout (设置 Configuration Client model 消息响应超时)	10
3.4.1.22 provision-adv (启动 PB-ADV 方式配网)	10
3.4.2 Configuration Client model	11
3.4.2.1 get-comp (读取目标设备一页成分数据)	11
3.4.2.2 beacon (secure network beacon 状态获取/设置)	11
3.4.2.3 beacon-listen (打印接收到的 unprovisioned Beacon)	11
3.4.2.4 ttl (节点 ttl 值获取/设置)	12
3.4.2.5 friend (节点 Friend 状态获取/设置)	12
3.4.2.6 gatt-proxy (节点 GATT Proxy 状态获取/设置)	12
3.4.2.7 relay (节点 relay 状态获取/设置)	13
3.4.2.8 net-key-add (添加 Netkey 到节点 Netkey List)	13
3.4.2.9 net-key-get (获取当前的 Netkey)	13
3.4.2.10 net-key-del (删除 Netkey)	14
3.4.2.11 net-key-update (更新目标节点的一个 Netkey)	14
3.4.2.12 krp-get (获取目标节点密钥刷新阶段)	14
3.4.2.13 krp-set (配置目标节点密钥刷新阶段)	14
3.4.2.14 app-key-add (添加 AppKey 到节点 Appkey List)	15
3.4.2.15 app-key-del (删除 AppKey)	15
3.4.2.16 app-key-get (获取 AppKey)	15
3.4.2.17 net-transmit-param (配置/获取网络传输参数)	16
3.4.2.18 mod-app-bind (给特定模型绑定指定 AppKey)	16
3.4.2.19 mod-app-get (获取跟 Model 绑定的 AppKey)	16
3.4.2.20 mod-app-unbind (移除模型绑定的 AppKey)	17
3.4.2.21 mod-pub (获取/设置节点模型的发布地址和参数)	17
3.4.2.22 mod-sub-add (添加模型订阅)	17

3.4.2.23 mod-sub-del (删除模型订阅)	18
3.4.2.24 mod-sub-add-va (添加虚拟地址到模型的订阅列表)	18
3.4.2.25 mod-sub-del-va (从模型的订阅列表中删除虚拟地址)	19
3.4.2.26 mod-sub-get (获取模型订阅列表)	19
3.4.2.27 hb-sub (获取/设置节点的 HeartBeat 订阅状态)	19
3.4.2.28 hb-pub (获取/设置 HeartBeat 广播配置)	20
3.4.3 Health Client model	20
3.4.3.1 fault-get (健康模型故障列表获取)	20
3.4.3.2 fault-clear (健康模型故障列表清空)	20
3.4.3.3 fault-clear-unack (健康模型故障列表无回复清空)	20
3.4.3.4 fault-test (健康模型元素自检)	21
3.4.3.5 fault-test-unack (健康模型元素无回复自检)	21
3.4.3.6 period-get (获取健康状态发布周期)	21
3.4.3.7 period-set (设置健康状态发布周期)	22
3.4.3.8 period-set-unack (无回复设置健康状态发布周期)	22
3.4.3.9 attention-get (获取 Attention Timer 状态)	22
3.4.3.10 attention-set (设置 Attention Timer 状态)	22
3.4.3.11 attention-set-unack (无回复设置 Attention Timer 状态)	23
3.4.4 Health Server model	23
3.4.4.1 add-fault (添加故障列表)	23
3.4.4.2 del-fault (删除故障列表)	23
3.4.5 网络数据库操作	24
3.4.5.1 cdb-create (创建 Mesh 网络数据库)	24
3.4.5.2 cdb-clear (清空 Mesh 网络数据库)	24
3.4.5.3 cdb-show (显示 Mesh 网络数据库)	24
3.4.5.4 cdb-node-add (添加节点到 Mesh 网络数据库)	24
3.4.5.5 cdb-node-del (删除 Mesh 网络数据库中节点)	25
3.4.5.6 cdb-subnet-add (添加子网到 Mesh 网络数据库)	25
3.4.5.7 cdb-subnet-del (删除 Mesh 网络数据库中子网)	25
3.4.5.8 cdb-app-key-add (添加 Appkey 到 Mesh 网络数据库)	26
3.4.5.9 cdb-app-key-del (删除 Mesh 网络数据库中 Appkey)	26
3.4.6 Generic OnOff Server	26
3.4.6.1 onoff-srv-set (设置本设备 on/off 服务端状态)	26
3.4.7 Generic OnOff Client	27

3.4.7.1 onoff-cli-get（客户端获取 on/off 状态）	27
3.4.7.2 onoff-cli-set（客户端配置 on/off 状态）	27
3.4.7.3 onoff-cli-set-unack（无回复客户端配置 on/off 状态）	27
4 示例说明	28
4.1 mesh 配网示例（PB-GATT）	28
4.1.1 示例简介	28
4.1.1.1 获取方式	28
4.1.1.2 准备工作	28
4.1.1.3 操作步骤	28
4.1.2 效果展示	28
4.2 mesh 配网示例（PB-ADV）	32
4.2.1 示例简介	32
4.2.1.1 获取方式	32
4.2.1.2 准备工作	32
4.2.1.3 操作步骤	33
4.2.2 效果展示	33

表格目录

表 2-1	Mesh 命令的文件位置.....	2
表 3-1	Mesh 命令配置列表.....	3
表 3-2	Mesh 命令语法约定.....	3
表 3-3	BLE 命令格式说明.....	3
表 3-4	Mesh 命令简要说明.....	4

1 前言

1.1 文档简介

本文档介绍了 XR806 SDK 的 Mesh 命令的用法及使用示例，以帮助用户使用 Mesh 命令快速上手 Mesh 功能开发。

1.2 目标读者

XR806 Mesh 功能测试、开发相关人员。

1.3 适用范围

此文档适用于 XR806 SDK，支持 XR806 系列芯片产品。

1.4 文档约定

1.4.1 标志说明

本文档采用各种醒目的标志来表示在操作过程中应该特别注意的地方，这些标志的含义如下：

标识	说明
 警告	该标志后的说明应给予格外关注，如果不遵守，可能会导致人员受伤或死亡。
 注意	提醒操作中应注意的事项。不当的操作可能会损坏器件，影响可靠性、降低性能等。
 说明	为准确理解文中指令、正确实施操作而提供的补充或强调信息。
 窍门	一些容易忽视的小功能、技巧。了解这些功能或技巧能帮助解决特定问题或者节省操作时间。

2 概述

2.1 背景说明

为了便于 Mesh 功能的测试、开发，XR806 SDK 集成了 Zephyr 项目的 Bluetooth Mesh 命令。SDK 中的 Mesh 命令主要封装在 cmd_mesh.c 文件中，覆盖了 Mesh 常用的功能场景。

2.2 规格特性

XR806 SDK 的 Bluetooth Mesh 命令支持节点（PB-GATT 及 PB-ADV 方式均支持）、配网者、配置模型、健康模型、通用开关模型等基本操作，具体命令的使用请参考“命令说明”小节。

2.3 文件位置

以 SDK 包为根目录，Mesh 命令涉及到的主要文件位置如下。

表 2-1 Mesh 命令的文件位置

组件名	文件分类	文件位置
Mesh 命令	源码文件	./project/common/cmd/cmd_mesh.c
	示例工程	./project/demo/bluetooth/ble_demo/

3 应用说明

3.1 应用简述

Mesh 命令已经集成到 XR806 SDK，下面将对 Mesh 命令的使用进行介绍。

3.2 配置说明

使用 Mesh 命令功能需要先将工程配置中的 Mesh 功能项打开。

表 3-1 Mesh 命令配置列表

配置项	配置说明
Mesh 功能使能	设置说明： 此项配置用于在 SDK 中启用 Mesh 功能。 设置方式： 在控制台执行命令“make menuconfig”，将以下功能项打开： [*] BLE Controller [*] Ble Host [*] Bluetooth Mesh support 建议参考 ble_demo 的工程配置 defconfig。

3.3 格式说明

Mesh 命令的语法定约如下表。

表 3-2 Mesh 命令语法定约

符号	说明
=	参数引导符
,	参数分隔符
<>	强制使用
[]	选择使用

在上述语法定约下，Mesh 命令有以下格式定义。

表 3-3 BLE 命令格式说明

分类	格式	说明
命令格式	mesh <CMD> [[p1],[p2],[p3[=]],...]	mesh：命令前缀，可改变 <CMD>：命令字符 [p1]/[p2]/[p3]：参数
响应格式	<ACK> <status-code> <respond-description>	<ACK>：命令响应前缀，固定。

分类	格式	说明
		<status-code>：状态码，为 200 时表示执行成功，为 4xx 时表示执行出错。 <respond-description>：响应消息。

3.4 命令说明

Mesh 命令提供了节点、配网者、模型操作等命令，命令的简要说明如下表所示。

表 3-4 Mesh 命令简要说明

命令	简要说明
通用配置命令	主要包含了模块初始化、广播、配网等命令，使用 mesh 命令必须先执行 BLE 初始化和 mesh 初始化。
Configuration Client model 操作命令	基础模型中客户端的配置模型操作命令。
Health Client model 操作命令	基础模型中客户端的健康模型操作命令。
Health Server model 操作命令	基础模型中服务端的健康模型操作命令。
网络数据库操作命令	提供了 Mesh 网络数据库配置等命令。
Generic OnOff Server 操作命令	通用模型中服务端 OnOff 模型操作命令。
Generic OnOff Client 操作命令	通用模型中客户端 OnOff 模型操作命令。

3.4.1 通用配置命令

3.4.1.1 init (mesh初始化)

信息项	说明
原型	mesh init
功能	mesh 模块初始化
参数	N/A
响应	<ACK> 200 OK
示例	N/A
备注	若系统启动后没有进行 BLE 初始化，执行 mesh 命令前都需要先执行 BLE 初始化命令。

3.4.1.2 reset (重置mesh节点)

信息项	说明
原型	mesh reset <addr>
功能	重置 mesh 节点
参数	<addr>：要重置的节点地址，可以是十进制或十六进制格式输入。
响应	<ACK> 200 OK

信息项	说明
示例	N/A
备注	重置后，节点将移出网络。

3.4.1.3 lpn（启用/关闭节点低功耗功能）

信息项	说明
原型	mesh lpn <value: off, on>
功能	启用/关闭节点低功耗功能
参数	<value>: off：关闭节点低功耗功能 on：启用节点低功耗功能
响应	<ACK> 200 OK
示例	启动节点低功耗模式：mesh lpn on
备注	低功耗节点与配网者角色只能二选一，启动后会自动扫描尝试与周围朋友节点建立朋友关系。朋友关系建立后会出现类型“Friendship established with xx”的打印。

3.4.1.4 poll（从朋友节点获取数据）

信息项	说明
原型	mesh poll
功能	从朋友节点获取数据
参数	N/A
响应	<ACK> 200 OK
示例	向朋友节点获取数据：mesh poll
备注	仅供低功耗节点使用，且与朋友节点建立朋友关系之后使用。

3.4.1.5 ident（启用节点身份认证广播）

信息项	说明
原型	mesh ident
功能	启用节点身份认证广播，该广播带有 proxy 的属性。
参数	N/A
响应	<ACK> 200 OK
示例	启用节点身份认证广播：mesh ident
备注	节点身份认证广播最长只能持续 60s，配网 60s 后若再需要扫描到该节点，需使用此命令。

3.4.1.6 dst (设置操作的节点对象)

信息项	说明
原型	mesh dst [destination address]
功能	想要操作的节点对象的地址
参数	[destination address]: 10 进制数, 长度 2 字节; 也可输入"local", 使用本地地址作为目的地址, 若不带参数则显示当前目的地址。
响应	<ACK> 200 OK
示例	设置目的地址为 1234: mesh dst 1234
备注	十进制输入, 回显结果为 16 进制, 如输入 1234, 回显显示 16 进制结果 0x4D2, 配网后使用。

3.4.1.7 netidx (设置mesh Netkey索引)

信息项	说明
原型	mesh netidx [NetIdx]
功能	设置 mesh Netkey 索引
参数	[NetIdx]: Netkey 的索引, 指使用哪个 Netkey, 每个 Netkey 对应了一个 net 或 subnet, 10 进制数据输入, 最大值为 65535; 若不带参数则显示当前所使用 Netkey 索引号。
响应	<ACK> 200 OK
示例	设置 mesh Netkey 索引为 1: mesh netidx 1
备注	设置当前节点要使用的 NetIdx

3.4.1.8 appidx (设置mesh Appkey索引)

信息项	说明
原型	mesh appidx [AppIdx]
功能	设置 mesh Appkey 索引
参数	[AppIdx]: Appkey 索引号, 10 进制数据输入, 最大值为 65535; 若不带参数则显示当前所使用 Appkey 索引。
响应	<ACK> 200 OK
示例	设置 mesh Appkey 索引为 1: mesh appidx 1
备注	N/A

3.4.1.9 net-send (网络内发送数据)

信息项	说明
原型	mesh net-send <hex string>
功能	网络内发送数据。命令实现使用内部 API, 只用于测试。

信息项	说明
参数	<hex string>: 16 进制字符串, 最长 32 字节
响应	<ACK> 200 OK
示例	在网络内发送 ABCD: mesh net-send ABCD
备注	配网完成后使用, 可以在子网内进行自组消息发送, 可用于 Debug 等方面。

3.4.1.10 iv-update (切换IV更新状态)

信息项	说明
原型	mesh iv-update
功能	切换 IV 更新状态。命令实现使用内部 API, 只用于测试。
参数	N/A
响应	<ACK> 200 OK
示例	切换 IV 更新状态: mesh iv-update
备注	切换到 IV 进行更新模式后, 默认要 96 个小时才会自动进行 IV 更新。

3.4.1.11 iv-update-test (使能/关闭IV更新测试)

信息项	说明
原型	mesh iv-update-test <value: off, on>
功能	使能/关闭 IV 更新测试。命令实现使用内部 API, 只用于测试。
参数	<value> on: 使能 off: 关闭
响应	<ACK> 200 OK
示例	使能 IV 更新模式: mesh iv-update-test on
备注	在 iv update test 模式下, 可以立即进行 IV 更新功能。

3.4.1.12 rpl-clear (清空重传保护列表)

信息项	说明
原型	mesh rpl-clear
功能	清空重传保护列表。命令实现使用内部 API, 只用于测试。
参数	N/A
响应	<ACK> 200 OK
示例	清空重传保护列表: mesh rpl-clear
备注	重传列表是为了防重放攻击。

3.4.1.13 pb-gatt (PB-GATT广播)

信息项	说明
原型	mesh pb-gatt <val: off, on>
功能	PB-GATT 广播
参数	<val>: on: 打开广播 off: 关闭广播
响应	<ACK> 200 OK
示例	启动 PB-GATT 方式广播: mesh pb-gatt on
备注	用于 PB-GATT 方式等待配网, provisioner 此时可以通过 PB-GATT 方式搜到此节点并对此节点配网。

3.4.1.14 pb-adv (PB-ADV广播)

信息项	说明
原型	mesh pb-adv <val: off, on>
功能	PB_ADV 广播
参数	<val>: on: 打开广播 off: 关闭广播
响应	<ACK> 200 OK
示例	启动 PB-ADV 方式广播: mesh pb-adv on
备注	用于 PB-ADV 方式等待配网, provisioner 此时可以通过 PB-ADV 方式搜到此节点并对此节点配网。

3.4.1.15 uuid (设置mesh设备UUID)

信息项	说明
原型	mesh uuid <UUID: 1-16 hex values>
功能	设置 mesh 设备 UUID
参数	<UUID>: mesh 设备 UUID 号, 最长 128bit
响应	<ACK> 200 OK
示例	设置设备 UUID 为 dddc: mesh uuid dddc
备注	设备 UUID 设置为 dddc, 在包中实际情况为 00000000000000000000000000000000dcdd, 即小端排列, 不同节点设备在被配网前需配置不同 UUID, 若相同会造成节点之间冲突。

3.4.1.16 input-num (输入式OOB认证数字输入)

信息项	说明
原型	mesh input-num <number>
功能	输入式 OOB 认证数字输入
参数	<number>: 需要输入的密钥数字, 十进制数格式, 为纯数字, 输入最大长度为 6
响应	<ACK> 200 OK
示例	密钥输入 123456: mesh input-num 123456
备注	配网之前将配网方式设置为 input num, 则在启动配网后, 会提示需要输入数字。

3.4.1.17 input-str (输入式OOB认证字符串输入)

信息项	说明
原型	mesh input-str <string>
功能	输入式 OOB 认证字符串输入
参数	<string>: 需要输入的密钥字符串, 输入最大长度为 6
响应	<ACK> 200 OK
示例	密钥输入 ABCDEF: mesh input-str ABCDEF
备注	配网之前将配网方式设置为 input string, 则在启动配网后, 会提示需要输入字符串。

3.4.1.18 static-oob (静态OOB值设置)

信息项	说明
原型	mesh static-oob [val: 1-16 hex values]
功能	静态 OOB 值设置
参数	[val]: 16 进制数据, 最长 8 个 byte; 可不带参数, 不带参数时为清空静态 OOB 值。
响应	<ACK> 200 OK
示例	设置静态 OOB 值为 123456: mesh static-oob 123456
备注	N/A

3.4.1.19 prov-method (配网认证方式配置)

信息项	说明
原型	mesh prov-method <method: static_oob, input, output, none> [action: number, string]
功能	配网认证方式配置, 仅 Provisioner 使用
参数	<method> static_oob: 设置使用静态 OOB 配网方式 input: 设置使用输入 OOB 身份验证方式 output: 设置使用输出 OOB 身份验证方式

信息项	说明
	none：不使用 OOB 身份验证进行配网 [action]：未配网设备选用的配网认证方式 number：数字方式 string：字符串方式
响应	<ACK> 200 OK
示例	不使用 OOB 身份验证进行配网：mesh prov-method none
备注	N/A

3.4.1.20 provision（配网者将自身配入到网络）

信息项	说明
原型	mesh provision <NetKeyIndex> <addr> [IVIndex]
功能	用于配网者将自身配入到网络中，形成一个 mesh 网络。
参数	<NetKeyIndex>：NetKey 索引，10 进制 <addr>：10 进制数据，不可配置为 0 [IVIndex]：IV 索引
响应	<ACK> 200 OK
示例	创建一个子网：mesh provision 0 1 0
备注	配置本地设备到 mesh 网络中，一般用于配网者将自身配入 mesh 网络中。

3.4.1.21 timeout（设置Configuration Client model消息响应超时）

信息项	说明
原型	mesh timeout [timeout in seconds]
功能	设置 Configuration Client model 消息响应超时
参数	[timeout in seconds]：超时时间设置，单位为 s
响应	<ACK> 200 OK
示例	设置 Configuration Client model 消息响应超时为 3s：mesh timeout 3
备注	默认 timeout 时间为 2s，消息发送后需要进行等待。

3.4.1.22 provision-adv（启动PB-ADV方式配网）

信息项	说明
原型	mesh provision-adv <UUID> <NetKeyIndex> <addr> <AttentionDuration>
功能	启动 PB-ADV 方式配网，配网者通过该方式可以将其他节点配置到网络去。
参数	<UUID>：Beacon 信标中的 UUID，为了区分设备，16 进制，128bit。 <NetKeyIndex>：配网所用 NetKey 索引，10 进制。

信息项	说明
	<addr>: 分配给节点的地址, 10 进制数据。 <AttentionDuration>: 配网持续最长时间, 单位为 s, 10 进制。
响应	<ACK> 200 OK
示例	N/A
备注	该命令需要使能 CONFIG_BT_MESH_PROVISIONER 时有效

3.4.2 Configuration Client model

3.4.2.1 get-comp (读取目标设备一页成分数据)

信息项	说明
原型	mesh get-comp [page]
功能	读取目标设备一页成分数据
参数	[page]: 需要读取的页
响应	<ACK> 200 OK
示例	读取目标第一页成分数据: mesh get-comp 0
备注	N/A

3.4.2.2 beacon (secure network beacon状态获取/设置)

信息项	说明
原型	mesh beacon [val: off, on]
功能	secure network beacon 状态获取/设置
参数	[val] on: 设置节点广播 secure network beacon off: 设置节点不广播 secure network beacon
响应	<ACK> 200 OK
示例	设置 secure network beacon 状态: mesh beacon on
备注	可不带参数, 不带参数获取 secure network beacon 状态信息

3.4.2.3 beacon-listen (打印接收到的unprovisioned Beacon)

信息项	说明
原型	mesh beacon-listen <val: off, on>
功能	打印接收到的 unprovisioned Beacon
参数	[val] on: 打印打开 off: 打印关闭

信息项	说明
响应	<ACK> 200 OK
示例	打印接收到的 unprovisioned beacon： mesh beacon-listen on
备注	一般为配网者使用

3.4.2.4 ttl（节点ttl值获取/设置）

信息项	说明
原型	mesh ttl [ttl: 0x00, 0x02-0x7f]
功能	节点 ttl 值获取/设置
参数	[ttl]：需要设置的 ttl 值，10 进制数据，设置范围应该为 0x0 或 0x2~0x7F，不能设置为 0x01 或 0x80~0xFF。
响应	<ACK> 200 OK
示例	设置节点 ttl 为 7： mesh ttl 7
备注	可不带参数，不带参数获取节点当前 ttl 值。

3.4.2.5 friend（节点Friend状态获取/设置）

信息项	说明
原型	mesh friend [val: off, on]
功能	开启/关闭节点 Friend 功能
参数	[val] on：开启 off：关闭
响应	<ACK> 200 OK
示例	设置节点 Friend 功能开启： mesh friend on
备注	可不带参数，不带参数获取节点当前 Friend 状态。

3.4.2.6 gatt-proxy（节点GATT Proxy状态获取/设置）

信息项	说明
原型	mesh gatt-proxy [val: off, on]
功能	节点 GATT Proxy 状态获取/设置
参数	[val] on：开启 off：关闭
响应	<ACK> 200 OK
示例	设置节点 Proxy 功能开启： mesh gatt-proxy on
备注	可不带参数，不带参数获取节点当前 GATT Proxy 状态。

3.4.2.7 relay（节点relay状态获取/设置）

信息项	说明
原型	mesh relay [<val: off, on> [<count: 0-7> [interval: 10-320]]]
功能	节点 relay 状态获取/设置
参数	val on：开启 off：关闭 count：中继次数，最大值为 7，可不输入，默认值为 2。 interval：中继间隔，输入值为 10 的倍数，单位 ms，最大值为 32，即 320ms，最小值要大于 0，可不输入，默认值为 20。
响应	<ACK> 200 OK
示例	设置 relay 特性开启，中继 3 次，间隔 30ms：mesh relay on 3 3
备注	可不带参数，不带参数获取节点当前 Relay 状态。

3.4.2.8 net-key-add（添加Netkey到节点Netkey List）

信息项	说明
原型	mesh net-key-add <NetKeyIndex> [val]
功能	添加 Netkey 到节点 Netkey List
参数	<NetKeyIndex>：需要添加的 NetKey 索引 [val]：Netkey 值，16 进制数据，最长为 16 字节；val 值可不添加，不添加则使用系统默认 key 值。
响应	<ACK> 200 OK
示例	添加索引为 1，值为 0x1234：mesh net-key-add 1 1234
备注	N/A

3.4.2.9 net-key-get（获取当前的Netkey）

信息项	说明
原型	mesh net-key-get
功能	获取当前的 Netkey
参数	N/A
响应	<ACK> 200 OK
示例	N/A
备注	N/A

3.4.2.10 net-key-del (删除Netkey)

信息项	说明
原型	mesh net-key-del <NetKeyIndex>
功能	删除 Netkey
参数	<NetKeyIndex>: 需要删除的 NetKey 索引
响应	<ACK> 200 OK
示例	删除索引为 1 的 Netkey: mesh net-key-del 1
备注	N/A

3.4.2.11 net-key-update (更新目标节点的一个NetKey)

信息项	说明
原型	mesh net-key-update <NetKeyIndex> [val]
功能	更新目标节点的一个 NetKey
参数	<NetKeyIndex>: 将要更新的 NetKey 的索引 [val]: 新的 NetKey 值, 可不配置该参数, 此时 NetKey 值将会随机产生。
响应	<ACK> 200 OK
示例	更新索引为 1 的 Netkey: mesh net-key-update
备注	若使能 CONFIG_BT_MESH_CDB, 新的 NetKey 值会从 CDB 中取。

3.4.2.12 krp-get (获取目标节点密钥刷新阶段)

信息项	说明
原型	mesh krp-get <NetKeyIndex>
功能	获取目标节点密钥刷新阶段
参数	<NetKeyIndex>: 目标节点的 NetKey 的索引
响应	<ACK> 200 OK
示例	N/A
备注	N/A

3.4.2.13 krp-set (配置目标节点密钥刷新阶段)

信息项	说明
原型	mesh krp-set <NetKeyIndex> <Transition:0-3>
功能	配置目标节点密钥刷新阶段
参数	<NetKeyIndex>: 目标节点的 NetKey 的索引 <Transition>: 密钥刷新阶段, 将要把目标节点密钥刷新阶段设置到的值, 可取值 0-3, 实际

信息项	说明
	上一般只会设 2 和 3
响应	<ACK> 200 OK
示例	配置目标节点 NetKey 的索引是 1 的密钥刷新阶段是 2：mesh krp-set 1 2
备注	N/A

3.4.2.14 app-key-add（添加AppKey到节点Appkey List）

信息项	说明
原型	mesh app-key-add <NetKeyIndex> <AppKeyIndex> [val]
功能	添加 AppKey 到节点 Appkey List
参数	<NetKeyIndex>：需要添加 AppKey 的子网的 NetKey 索引 <AppKeyIndex>：需要添加的 AppKey 索引 [val]：Appkey 值，16 进制数据，最长为 16 字节；val 值可不添加，不添加则使用系统默认 key 值。
响应	<ACK> 200 OK
示例	添加 netkey 索引为 1，appkey 索引为 1，appkey 值为 0x1234：mesh app-key-add 1 1 1234
备注	N/A

3.4.2.15 app-key-del（删除AppKey）

信息项	说明
原型	mesh app-key-del <NetKeyIndex> <AppKeyIndex>
功能	删除 AppKey
参数	<NetKeyIndex>：需要删除 AppKey 的子网的 NetKey 索引 <AppKeyIndex>：需要删除的 AppKey 索引
响应	<ACK> 200 OK
示例	删除 netkey 索引为 1，appkey 索引为 1 的 AppKey：mesh app-key-del 1 1
备注	N/A

3.4.2.16 app-key-get（获取AppKey）

信息项	说明
原型	mesh app-key-get <NetKeyIndex>
功能	获取 AppKey
参数	<NetKeyIndex>：需要获取的 AppKey 所对应的 NetKey 索引
响应	<ACK> 200 OK
示例	N/A

信息项	说明
备注	该命令将获取所有绑定至指定 NetKey 的 AppKey

3.4.2.17 net-transmit-param（配置/获取网络传输参数）

信息项	说明
原型	mesh net-transmit-param [<count: 0-7> <interval: 10-320>]
功能	配置/获取网络传输参数，可以不配置参数表示获取传输参数
参数	[count]: 传输数量 [interval]: 传输间隔
响应	<ACK> 200 OK
示例	N/A
备注	N/A

3.4.2.18 mod-app-bind（给特定模型绑定指定AppKey）

信息项	说明
原型	mesh mod-app-bind <addr> <AppIndex> <Model ID> [Company ID]
功能	给特定模型绑定指定 AppKey
参数	<addr>: 元素的 unicast 地址 <AppIndex>: 指定的 App Key 索引 <Model ID>: 模型标识符 [Company ID]: 厂商标识符，可不添加，使用默认值 0xFFFF
响应	<ACK> 200 OK
示例	将 element 1 对应 model 2 与 AppIndex 3 绑定: mesh mod-app-bind 1 3 2
备注	N/A

3.4.2.19 mod-app-get（获取跟Model绑定的AppKey）

信息项	说明
原型	mesh mod-app-get <elem addr> <Model ID> [Company ID]
功能	获取跟 Model 绑定的 AppKey
参数	<elem addr>: 模型所在的元素地址 <Model ID>: 模型标识符 [Company ID]: 厂商标识符
响应	<ACK> 200 OK
示例	N/A
备注	若配置 Company ID 参数则表示获取跟厂商模型绑定的 AppKey

3.4.2.20 mod-app-unbind（移除模型绑定的AppKey）

信息项	说明
原型	mesh mod-app-unbind <addr> <AppIndex> <Model ID> [Company ID]
功能	移除指定模型的 AppKey
参数	<addr>：元素的 unicast 地址 <AppIndex>：AppKey 索引 <Model ID>：模型标识符 [Company ID]：厂商标识符
响应	<ACK> 200 OK
示例	N/A
备注	若配置 Company ID 参数则表示移除跟厂商模型绑定的 AppKey

3.4.2.21 mod-pub（获取/设置节点模型的发布地址和参数）

信息项	说明
原型	mesh mod-pub <addr> <mod id> [cid] [<PubAddr> <AppKeyIndex> <cred: off, on> <ttl> <period> <count> <interval>]
功能	获取/设置节点模型的发布地址和参数
参数	<addr>：元素的 unicast 地址 <mod id>：模型标识符 ID [cid]：company id，默认值为 0xFFFF <PubAddr>：发布地址 <AppKeyIndex>：指定的 AppKey 索引 <cred>：friendship credential flag，低功耗节点使用，是否在朋友关系之间使用朋友加密密钥。 <ttl>：需要设置的 ttl 值，10 进制数据，设置范围应该为 0x0 或 0x2~0x7F，不能设置为 0x01，0x80~0xFF。 <period>：发布周期 <count>：重发次数，最大重发次数为 7 <interval>：重发间隔，单位为 50ms，取值范围为 1~31
响应	<ACK> 200 OK
示例	获取节点的模型的发布地址：mesh mod-pub 1 1
备注	发布参数配置最少需要 9 个参数，发布参数获取最少需要 2 个参数，cid 参数可选择性带或不带，若 cid 参数不带，则使用默认值 0xFFFF。

3.4.2.22 mod-sub-add（添加模型订阅）

信息项	说明
原型	mesh mod-sub-add <elem addr> <sub addr> <Model ID> [Company ID]

信息项	说明
功能	向模型订阅列表新增一条模型订阅地址
参数	<code><elem addr></code> : 元素的 unicast 地址, 以 10 进制数据输入 <code><sub addr></code> : 需要添加到订阅列表的地址, 以 10 进制数据输入 <code><Model ID></code> : 模型 ID, 以 10 进制数据输入 <code>[Company ID]</code> : 可不带, 默认值为 0xFFFF, 以 10 进制数据输入
响应	<code><ACK> 200 OK</code>
示例	添加订阅地址 2 到元素 3 的模型 4 中: <code>mesh mod-sub-add 3 2 4</code>
备注	N/A

3.4.2.23 mod-sub-del (删除模型订阅)

信息项	说明
原型	<code>mesh mod-sub-del <elem addr> <sub addr> <Model ID> [Company ID]</code>
功能	用于从模型订阅列表中删除特定模型的特定订阅地址
参数	<code><elem addr></code> : 元素的 unicast 地址, 以 10 进制数据输入 <code><sub addr></code> : 需要删除的订阅地址, 以 10 进制数据输入 <code><Model ID></code> : 模型 ID, 以 10 进制数据输入 <code>[Company ID]</code> : 可不带, 默认值为 0xFFFF, 以 10 进制数据输入
响应	<code><ACK> 200 OK</code>
示例	删除元素 3 的模型 4 中的订阅地址 2: <code>mesh mod-sub-del 3 2 4</code>
备注	N/A

3.4.2.24 mod-sub-add-va (添加虚拟地址到模型的订阅列表)

信息项	说明
原型	<code>mesh mod-sub-add-va <elem addr> <Label UUID> <Model ID> [Company ID]</code>
功能	添加虚拟地址到模型的订阅列表
参数	<code><elem addr></code> : 元素的 unicast 地址, 以 10 进制数据输入 <code><Label UUID></code> : 需要添加的 label UUID, 以 10 进制数据输入 <code><Model ID></code> : 模型 ID, 以 10 进制数据输入 <code>[Company ID]</code> : 可不带, 默认值为 0xFFFF, 以 10 进制数据输入
响应	<code><ACK> 200 OK</code>
示例	添加虚拟地址 2 到元素 3 的模型 4 中: <code>mesh mod-sub-add-va 3 2 4</code>
备注	N/A

3.4.2.25 mod-sub-del-va（从模型的订阅列表中删除虚拟地址）

信息项	说明
原型	mesh mod-sub-del-va <elem addr> <Label UUID> <Model ID> [Company ID]
功能	从模型的订阅列表中删除虚拟地址
参数	<elem addr>：元素的 unicast 地址，以 10 进制数据输入 <Label UUID>：需要删除的 label UUID，以 10 进制数据输入 <Model ID>：模型 ID，以 10 进制数据输入 [Company ID]：可不带，默认值为 0xFFFF，以 10 进制数据输入
响应	<ACK> 200 OK
示例	删除元素 3 的模型 4 中的虚拟地址 2：mesh mod-sub-del-va 3 2 4
备注	N/A

3.4.2.26 mod-sub-get（获取模型订阅列表）

信息项	说明
原型	mesh mod-sub-get <elem addr> <Model ID> [Company ID]
功能	获取模型订阅列表
参数	<elem addr>：元素的 unicast 地址，以 10 进制数据输入 <Model ID>：模型 ID，以 10 进制数据输入 [Company ID]：可不带，默认值为 0xFFFF，以 10 进制数据输入
响应	<ACK> 200 OK
示例	获取元素 3 的模型 4 中的订阅列表：mesh mod-sub-get 3 4
备注	N/A

3.4.2.27 hb-sub（获取/设置节点的HeartBeat订阅状态）

信息项	说明
原型	mesh hb-sub [<src> <dst> <period>]
功能	获取/设置节点的 HeartBeat 订阅状态
参数	[src]：源地址，HeartBeat 的源发送地址，以 10 进制数据输入； [ds]：目的地址，HeartBeat 的发送地址，以 10 进制数据输入； [period]：HeartBeat 接收周期，以 10 进制数据输入，实际使用时会转化为指数形式，即 $(1 \ll (\text{period} - 1))$ 。
响应	<ACK> 200 OK
示例	设置节点 HeartBeat 订阅状态，源地址 1，目的地址 2，周期为 3：mesh hb-sub 1 2 3
备注	可不带参数，不带参数获取节点当前 HeartBeat 订阅状态

3.4.2.28 hb-pub（获取/设置HeartBeat广播配置）

信息项	说明
原型	mesh hb-pub [<dst> <count> <period> <ttl> <features> <NetKeyIndex>]
功能	获取/设置 HeartBeat 广播配置
参数	[dst]: 心跳广播的目的地址，以 10 进制数据输入； [count]: 心跳消息发送次数，以 10 进制数据输入； [period]: 心跳消息发送间隔，以 10 进制数据输入； [ttl]: 心跳消息转发次数，以 10 进制数据输入； [feature]: 心跳消息支持的功能，以 10 进制数据输入； [NetKeyIndex]: 心跳消息使用的 NetKey 索引，以 10 进制数据输入
响应	<ACK> 200 OK
示例	获取心跳广播配置：mesh hb-pub
备注	可不带参数，不带参数获取节点当前 HeartBeat 配置状态

3.4.3 Health Client model

3.4.3.1 fault-get（健康模型故障列表获取）

信息项	说明
原型	mesh fault-get <Company ID>
功能	获取 Health Server 模型中已发生的故障列表
参数	<Company ID>: CID，以 10 进制数据输入，不可省略
响应	<ACK> 200 OK
示例	获取 0xFFFF 故障列表：mesh fault-get 65535
备注	N/A

3.4.3.2 fault-clear（健康模型故障列表清空）

信息项	说明
原型	mesh fault-clear <Company ID>
功能	清空 Health Server 模型中已发生的故障列表
参数	<Company ID>: CID，以 10 进制数据输入，不可省略
响应	<ACK> 200 OK
示例	清空 0xFFFF 故障列表：mesh fault-clear 65535
备注	N/A

3.4.3.3 fault-clear-unack（健康模型故障列表无回复清空）

信息项	说明
-----	----

信息项	说明
原型	mesh fault-clear-unack <Company ID>
功能	无回复清空 Health Server 模型中已发生的故障列表
参数	<Company ID>: CID, 以 10 进制数据输入, 不可省略
响应	<ACK> 200 OK
示例	清空 0xFFFF 故障列表: mesh fault-clear-unack 65535
备注	N/A

3.4.3.4 fault-test (健康模型元素自检)

信息项	说明
原型	mesh fault-test <Company ID> <Test ID>
功能	Health Server 模型元素自检
参数	<Company ID>: CID, 以 10 进制数据输入, 不可省略 <Test ID>: 特殊自检流程 ID 号, 以 10 进制数据输入
响应	<ACK> 200 OK
示例	执行自检程序 1: mesh fault_test 65535 1
备注	自检完成后可能会改变 Health Server 模型中某些故障的状态

3.4.3.5 fault-test-unack (健康模型元素无回复自检)

信息项	说明
原型	mesh fault-test-unack <Company ID> <Test ID>
功能	Health Server 模型元素无回复自检
参数	<Company ID>: CID, 以 10 进制数据输入, 不可省略 <Test ID>: 特殊自检流程 ID 号, 以 10 进制数据输入
响应	<ACK> 200 OK
示例	执行自检程序 1: mesh fault-test-unack 65535 1
备注	N/A

3.4.3.6 period-get (获取健康状态发布周期)

信息项	说明
原型	mesh period-get
功能	获取健康状态发布周期
参数	N/A
响应	<ACK> 200 OK
示例	获取健康状态发布周期: mesh period-get

信息项	说明
备注	N/A

3.4.3.7 period-set（设置健康状态发布周期）

信息项	说明
原型	mesh period-set <divisor>
功能	设置健康状态发布周期
参数	<divisor>：需要设置的发布周期，以 10 进制数据输入，取值范围为 0~15
响应	<ACK> 200 OK
示例	设置健康状态发布周期：mesh period-set 2
备注	N/A

3.4.3.8 period-set-unack（无回复设置健康状态发布周期）

信息项	说明
原型	mesh period-set-unack <divisor>
功能	无回复设置健康状态发布周期
参数	<divisor>：需要设置的发布周期，以 10 进制数据输入，取值范围为 0~15
响应	<ACK> 200 OK
示例	设置健康状态发布周期：mesh period-set-unack 2
备注	N/A

3.4.3.9 attention-get（获取Attention Timer状态）

信息项	说明
原型	mesh attention-get
功能	获取 Attention Timer 状态
参数	N/A
响应	<ACK> 200 OK
示例	获取 Attention Timer 状态：mesh attention-get
备注	attention timer 在配网过程中使用

3.4.3.10 attention-set（设置Attention Timer状态）

信息项	说明
原型	mesh attention-set <timer>
功能	设置 Attention Timer 状态

信息项	说明
参数	<timer>：设置为非零时表示为 timer 启动，timer 值为 attention 持续时间，单位为 s，取值范围为 0~0xFF,以 10 进制数据输入。
响应	<ACK> 200 OK
示例	设置 Attention Timer 为开，时间为 5s：mesh attention-set 5
备注	N/A

3.4.3.11 attention-set-unack（无回复设置Attention Timer状态）

信息项	说明
原型	mesh attention-set-unack <timer>
功能	无回复设置 Attention Timer 状态
参数	<timer>：设置为非零时表示为 timer 启动，timer 值为 attention 持续时间，单位为 s，取值范围为 0~0xFF,以 10 进制数据输入。
响应	<ACK> 200 OK
示例	设置 Attention Timer 为开，时间为 5s：mesh attention-set-unack 5
备注	N/A

3.4.4 Health Server model

3.4.4.1 add-fault（添加故障列表）

信息项	说明
原型	mesh add-fault <Fault ID>
功能	添加故障列表
参数	<Fault ID>：添加的故障列表 ID，以 10 进制数据输入，不能为 0
响应	<ACK> 200 OK
示例	添加故障列表 1：mesh add-fault 1
备注	N/A

3.4.4.2 del-fault（删除故障列表）

信息项	说明
原型	mesh del-fault [Fault ID]
功能	删除故障列表
参数	<Fault ID>：添加的故障列表 ID，以 10 进制数据输入，不能为 0
响应	<ACK> 200 OK
示例	删除故障列表 1：mesh del-fault 1
备注	N/A

3.4.5 网络数据库操作

3.4.5.1 cdb-create（创建Mesh网络数据库）

信息项	说明
原型	mesh cdb-create [NetKey]
功能	创建 Mesh 网络数据库
参数	[NetKey]：网络密钥，16 进制数据输入，长度为 16 字节
响应	<ACK> 200 OK
示例	创建 Mesh 网络数据库：mesh cdb-create dddd
备注	N/A

3.4.5.2 cdb-clear（清空Mesh网络数据库）

信息项	说明
原型	mesh cdb-clear
功能	清空 Mesh 网络数据库
参数	N/A
响应	<ACK> 200 OK
示例	清空 Mesh 网络数据库：mesh cdb-clear
备注	N/A

3.4.5.3 cdb-show（显示Mesh网络数据库）

信息项	说明
原型	mesh cdb-show
功能	显示 Mesh 网络数据库
参数	N/A
响应	<ACK> 200 OK
示例	显示 Mesh 网络数据库所有数据：mesh cdb-show
备注	N/A

3.4.5.4 cdb-node-add（添加节点到Mesh网络数据库）

信息项	说明
原型	mesh cdb-node-add <UUID> <addr> <num-elem> <NetKeyIdx> [DevKey]
功能	添加节点到 Mesh 网络数据库
参数	<UUID>：节点 UUID，16 进制输入，长度 16 字节

信息项	说明
	<code><addr></code> : 节点地址, 10 进制输入 <code><num-elem></code> : 节点元素数量, 10 进制输入 <code><NetKeyId></code> : NetKey 索引, 10 进制输入 <code>[DevKey]</code> : 节点设备密钥, 16 进制输入, 长度 16 字节; 可不带该参数, 由系统产生随机数填充
响应	<code><ACK> 200 OK</code>
示例	添加节点到 Mesh 网络数据库: <code>mesh cdb-node-add 1234 2 3 4</code>
备注	N/A

3.4.5.5 cdb-node-del (删除Mesh网络数据库中节点)

信息项	说明
原型	<code>mesh cdb-node-del <addr></code>
功能	删除 Mesh 网络数据库中节点
参数	<code><addr></code> : 节点地址, 10 进制输入
响应	<code><ACK> 200 OK</code>
示例	删除 Mesh 网络数据库中节点 1: <code>mesh cdb-node-del 1</code>
备注	N/A

3.4.5.6 cdb-subnet-add (添加子网到Mesh网络数据库)

信息项	说明
原型	<code>mesh cdb-subnet-add <NetKeyId> [<NetKey>]</code>
功能	添加子网到 Mesh 网络数据库
参数	<code><NetKeyId></code> : NetKey 索引, 10 进制输入 <code>[NetKey]</code> : 网络密钥, 16 进制输入, 长度 16 字节; 可不带该参数, 由系统产生随机数填充
响应	<code><ACK> 200 OK</code>
示例	添加子网 3 到 Mesh 网络数据库: <code>mesh cdb-subnet-add 3</code>
备注	N/A

3.4.5.7 cdb-subnet-del (删除Mesh网络数据库中子网)

信息项	说明
原型	<code>mesh cdb-subnet-del <NetKeyId></code>
功能	删除 Mesh 网络数据库中子网
参数	<code><NetKeyId></code> : NetKey 索引, 10 进制输入
响应	<code><ACK> 200 OK</code>

信息项	说明
示例	删除 Mesh 网络数据库中子网 1：mesh cdb-subnet-del 1
备注	N/A

3.4.5.8 cdb-app-key-add（添加Appkey到Mesh网络数据库）

信息项	说明
原型	mesh cdb-app-key-add <NetKeyId> <AppKeyId> [<AppKey>]
功能	添加 Appkey 到 Mesh 网络数据库
参数	<NetKeyId>：NetKey 索引，10 进制输入 <AppKeyId>：AppKey 索引，10 进制输入 [AppKey]：App 密钥，16 进制输入，长度 16 字节；可不带该参数，由系统产生随机数填充
响应	<ACK> 200 OK
示例	添加 netkey2 的 appkey3 到 Mesh 网络数据库：mesh mesh cdb-app-key-add 2 3
备注	N/A

3.4.5.9 cdb-app-key-del（删除Mesh网络数据库中Appkey）

信息项	说明
原型	mesh cdb-app-key-del <AppKeyId>
功能	删除 Mesh 网络数据库中 Appkey
参数	<AppKeyId>：AppKey 索引，10 进制输入
响应	<ACK> 200 OK
示例	删除 Mesh 网络数据库中 appkey 1：mesh cdb-app-key-del 1
备注	N/A

3.4.6 Generic OnOff Server

3.4.6.1 onoff-srv-set（设置本设备on/off服务端状态）

信息项	说明
原型	mesh onoff-srv-set <val: off, on> [trans=<ms>] [delay=<ms>]
功能	设置本设备 on/off 服务端状态
参数	<val>：on 为开，off 为关 [trans]：渐变时间，即设备从一个状态变换到另一个状态所需要的时间，单位 ms [delay]：延迟时间，即设备从接收到消息到开始执行变化的时间，单位 ms
响应	<ACK> 200 OK
示例	设置本设备 onoff 状态为 off：mesh onoff-srv-set off
备注	N/A

3.4.7 Generic OnOff Client

3.4.7.1 onoff-cli-get（客户端获取on/off状态）

信息项	说明
原型	mesh onoff-cli-get
功能	获取服务端 on/off 状态
参数	N/A
响应	<ACK> 200 OK
示例	N/A
备注	N/A

3.4.7.2 onoff-cli-set（客户端配置on/off状态）

信息项	说明
原型	mesh onoff-cli-set <val: off, on> [trans=<ms>] [delay=<ms>]
功能	客户端去设置服务端 on/off 状态
参数	<val>: on 为开, off 为关 [trans]: 渐变时间, 即设备从一个状态变换到另一个状态所需要的时间, 单位 ms [delay]: 延迟时间, 即设备从接收到消息到开始执行变化的时间, 单位 ms
响应	<ACK> 200 OK
示例	配置服务端状态为 on: mesh onoff-cli-set on
备注	N/A

3.4.7.3 onoff-cli-set-unack（无回复客户端配置on/off状态）

信息项	说明
原型	mesh onoff-cli-set-unack <val: off, on> [trans=<ms>] [delay=<ms>]
功能	无回复客户端去设置服务端 on/off 状态
参数	<val>: on 为开, off 为关 [trans]: 渐变时间, 即设备从一个状态变换到另一个状态所需要的时间, 单位 ms [delay]: 延迟时间, 即设备从接收到消息到开始执行变化的时间, 单位 ms
响应	<ACK> 200 OK
示例	配置服务端状态为 on: mesh onoff-cli-set-unack on
备注	N/A

4 示例说明

本章节提供了两个示例，分别以 PB-GATT 和 PB-ADV 方式对 Mesh 的配网场景进行演示说明。

4.1 mesh 配网示例（PB-GATT）

4.1.1 示例简介

本示例通过演示移动设备给 Mesh 节点配网场景，简要介绍 mesh 命令的基本使用方法。

4.1.1.1 获取方式

Mesh 命令集成在 cmd_mesh.c 文件中,通过 ble_demo 工程可以完成 mesh 节点配网场景演示,ble_demo 工程位于 XR806 SDK 的/project/demo/bluetooth/ble_demo 目录。

4.1.1.2 准备工作

ble_demo 工程的硬件准备有如下。

1. 评估板：运行示例工程代码。
2. 串口线：连接评估板的 Uart0 插针，用于 console 控制台的输入输出。
3. PC 机：用于镜像烧录和 console 控制的输入输出。
4. 手机/平板：用于运行 nRF Mesh 软件，本文档的 nRF Mesh 采用的版本是 V2.4.1。

4.1.1.3 操作步骤

1. 编译 ble_demo 工程，完成烧写后，复位即可，示例代码自动运行。
2. 手机/平板安装并运行 nRF Mesh 软件。

4.1.2 效果展示

1. 在评估板中运行 ble_demo 工程后，在控制台中会打印出系统启动信息。

```
use default flash chip mJedec 0x0
[FD I]: mode: 0x10, freq: 96000000Hz, drv: 0
[FD I]: jedec: 0x0, suspend_support: 1
mode select:e
[VFS INF] LittleFS mount success.

platform information =====
XR806 SDK v1.0.0  Apr  2 2021 18:48:49

heap space [0x20d7ac, 0x247c00), size 238676

cpu  clock 160000000 Hz
HF   clock  40000000 Hz

sdk option:
```

```
XIP           : enable
INT LF OSC    : enable
SIP flash     : enable
```

```
=====
```

```
$>Welcome to Bluetooth Console Tools .....
$>
```

2. 进行 BLE 初始化，往串口输入 BLE 初始化命令：

```
$>ble init
```

成功初始化后，可以看到以下打印信息：

```
...
Device A not active,waking up
ble rf_init done!
BLE INIT ALL DONE!
BT Coex. Init. OK
== XRadio BLE HOST V2.5.0 ==
[bt] [WRN] : Controller to host flow control not supported
[bt] [INF] : No ID address. App must call settings_load()
[cmd] Bluetooth initialized
[bt] [INF] : Identity: CE:BA:E1:F6:4F:4A (random)
[bt] [INF] : HCI: version 5.0 (0x09) revision 0x0111, manufacturer 0x063d
[bt] [INF] : LMP: version 5.0 (0x09) subver 0x0111
[cmd] Settings Loaded
<ACK> 200 OK
```

3. 进行 Mesh 模块初始化，往串口输入命令：

```
$>mesh init
```

成功执行后有如下提示：

```
== XRadio BLE HOST V2.5.0 ==
[cmd] Mesh initialized
[cmd] Use "pb-adv on" or "pb-gatt on" to enable advertising
<ACK> 200 OK
```

4. 选用 PB-GATT 的承载方式：

```
$>mesh pb-gatt on
```

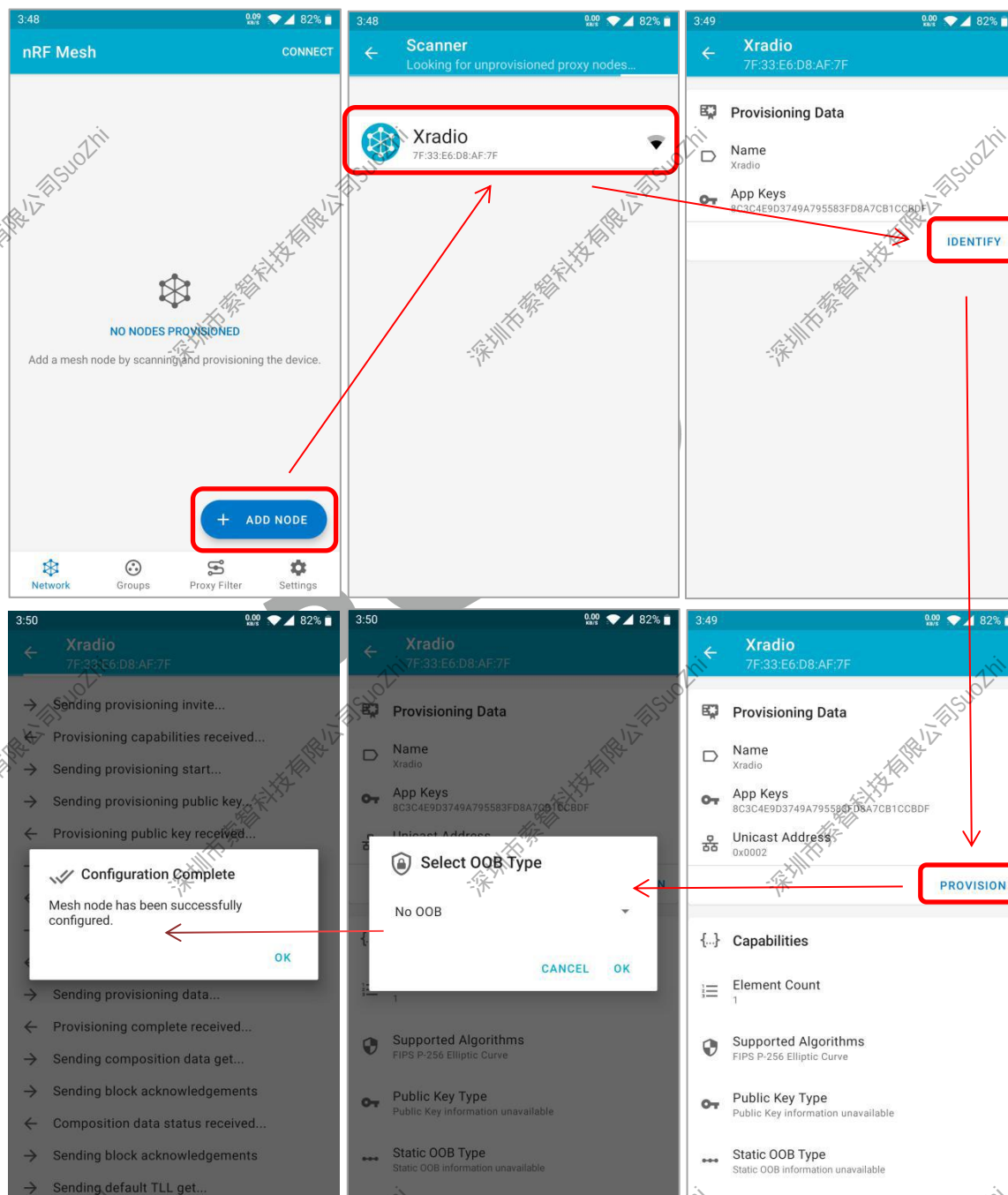
成功广播后有如下提示：

```
[bt] [INF] : Device UUID: 00000000-0000-0000-0000-00000000dddd
[cmd] PB-GATT enabled
```

```
<ACK> 200 OK
$ *****
[RandomAddress 7F:33:E6:D8:AF:7F ]
*****
```

5. 打开 nRF Mesh 软件，点击“ADD NODE”按钮，进行周围设备扫描，可以找到评估板的设备信息，即下图“Xradio”，点击该项，出现“IDENTIFY”页面，再点击“IDENTIFY”出现配网界面，然后点击“PROVISION”出现配网界面，选择 OBB 类型，开始进行配网。

图 4-1 nRF Mesh 配网



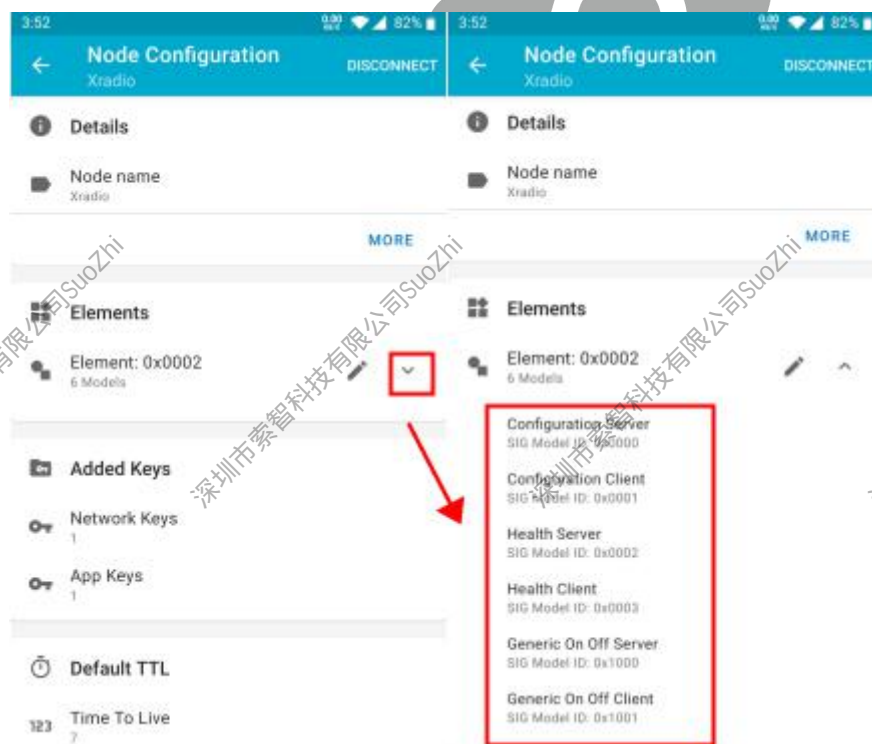
6. 配网成功后，可以看到 nRF Mesh 网络列表看到该节点的信息，如下图。

图 4-2 nRF Mesh 网络列表



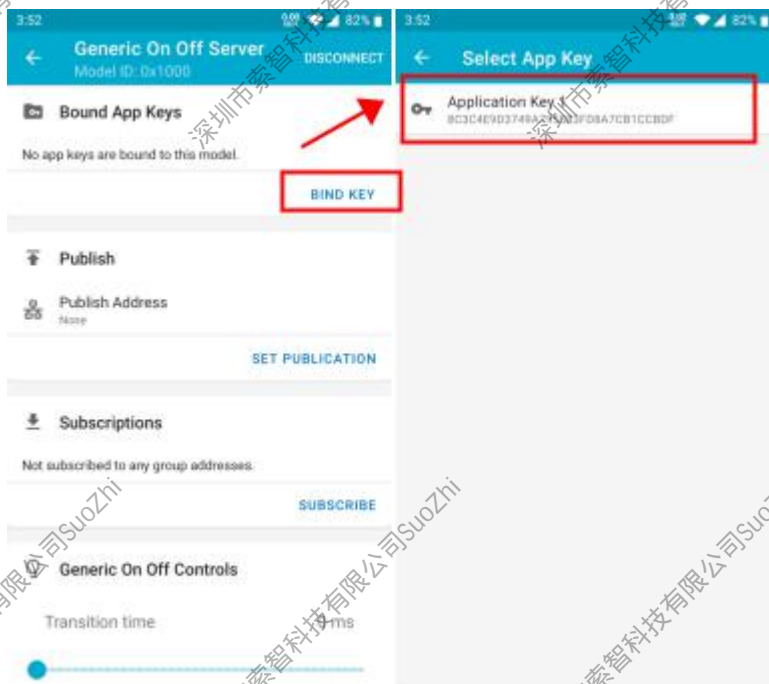
7. 点击红色方框，进入节点配置页面，可以看到节点的元素、模型等信息。

图 4-3 节点配置界面



8. 选择“Genenic On Off Server”模型进行操作：点击该模型选项，进入模型操作界面，如下图。进行模型操作需要先进行“APP Key”绑定，点击“BIND KEY”，然后选择 App Key。

图 4-4 模型操作界面



9. 进行“Generic On Off”：点击“ON”，可以看到设备的 On/ Off 状态开了：

```
[cmd] onoff get(0)
[cmd] onoff set(1)
[cmd] onoff get(1)
```

说明：Generic On Off 操作会先去获取 OnOff 状态，然后再配置，配置完成后再获取一次状态。

点击“OFF”，可以看到设备的 On/ Off 状态关闭了：

```
[cmd] onoff get(1)
[cmd] onoff set(0)
[cmd] onoff get(0)
```

4.2 mesh 配网示例（PB-ADV）

4.2.1 示例简介

本示例通过演示 mesh 通过 PB-ADV 方式给节点配网场景，简要介绍 Mesh 命令的基本使用方法。

4.2.1.1 获取方式

Mesh 命令集成在 cmd_mesh.c 文件中,通过 ble_demo 工程可以完成 mesh 节点配网场景演示,ble_demo 工程位于 XR806 SDK 的/project/demo/bluetooth/ble_demo 目录。

4.2.1.2 准备工作

ble_demo 工程的硬件准备有如下。

1. 评估板：2 块，运行示例工程代码。
2. 串口线：连接评估板的 Uart0 插针，用于 console 控制台的输入输出。

3. PC 机：用于镜像烧录和 console 控制的输入输出。

4.2.1.3 操作步骤

1. 编译 ble_demo 工程，两块评估板完成烧写后，复位即可，示例代码自动运行。
2. 输入 mesh 命令，完成配网场景。

4.2.2 效果展示

1. 两块评估板分别运行 ble_demo 工程。两块评估板以下用 A/B 简称,A 作为 Provisioner,B 作为 Node。
2. 评估板 A/B 分别执行 BLE 初始化命令：

```
$>ble init
```

3. 评估板 A/B 分别执行 Mesh 初始化命令：

```
$>mesh init
```

4. 评估板 B 配置 UUID，往串口输入 UUID 配置命令：

```
$>mesh uuid 1234
```

5. 评估板 B 启动 PB-ADV 方式广播：

```
$>mesh pb-adv on
```

成功执行后，会显示以下打印：

```
[bt] [INF] : Device UUID: 00000000-0000-0000-0000-000000003412
*****
[RandomAddress 7D:6D:0E:F1:09:F0 ]
*****
[cmd] PB-ADV enabled
<ACK> 200 OK
```

6. 评估板 A 创建 Mesh 网络数据库：

```
$>mesh cdb-create
```

7. 评估板 A 将自身加入网络中：

```
$>mesh provision 0 1 0
```

成功执行后，显示以下打印：

```
[bt] [INF] : Primary Element: 0x0001
*****
[RandomAddress 49:0B:7E:78:0D:82 ]
*****
[cmd] Local node provisioned, net_idx 0x0000 address 0x0001
```

8. 评估板 A 添加 appkey：

```
$>mesh app-key-add 0 0
```

添加成功执行后会显示以下打印：

```
[cmd] AppKey added, NetKeyIndex 0x0000 AppKeyIndex 0x0000
<ACK> 200 OK
```

9. 评估板 A 对本地 0x1000、0x1001 模型进行绑定：

```
$ mesh mod-app-bind 1 0 0x1000
[cmd] AppKey successfully bound
<ACK> 200 OK
```

```
$
$ mesh mod-app-bind 1 0 0x1001
[cmd] AppKey successfully bound
<ACK> 200 OK
```

添加成功执行后会显示以下打印：

```
[cmd] AppKey added, NetKeyIndex 0x0000 AppKeyIndex 0x0000
<ACK> 200 OK
```

10. 评估板 A 将评估板 B 加入网络中：

```
$>mesh provision-adv 1234 0 2 0
```

配网成功后可以看到以下打印：

```
[cmd] Provisioning link opened on PB-ADV
[cmd] Node provisioned, net_idx 0x0000 address 0x0002 elements 1
[cmd] Provisioning link closed on PB-ADV
```

同时评估板 B 也有以下配网完成打印：

```
Provisioning link opened on PB-ADV
[bt] [INF] : Primary Element: 0x0002
[cmd] Local node provisioned, net_idx 0x0000 address 0x0002
[cmd] Provisioning link closed on PB-ADV
```

11. 评估板 A 给评估板 B 添加 appkey：

```
$>mesh app-key-add 0 0
[cmd] AppKey added, NetKeyIndex 0x0000 AppKeyIndex 0x0000
<ACK> 200 OK
```

12. 评估板 A 对评估板 B 的 0x1000、0x1001 模型进行绑定：

```
$ mesh mod-app-bind 2 0 0x1000
[cmd] AppKey successfully bound
<ACK> 200 OK
```

```
$
$ mesh mod-app-bind 2 0 0x1001
[cmd] AppKey successfully bound
<ACK> 200 OK
```

13. 评估板 A 对评估板 B 进行“Genenic On Off Server”模型进行操作：在评估板 A 控制台输入“mesh onoff-cli-set on”命令，可以看到评估板 B 的 Genenic On Off Server 状态是打开了：

```
[cmd] onoff get(0)
[cmd] onoff set(1)
[cmd] onoff get(1)
```

输入“mesh onoff-cli-set off”命令，可以看到评估板 B 的 Genenic On Off Server 状态是关闭了：

```
[cmd] onoff get(1)
[cmd] onoff set(0)
[cmd] onoff get(0)
```

著作权声明

版权所有©2020 广州芯之联科技有限公司。保留一切权利。

本文档及内容受著作权法保护，其著作权由广州芯之联科技有限公司（“芯之联”）拥有并保留一切权利。

本文档是芯之联的原创作品和版权财产，未经芯之联书面许可，任何单位和个人不得擅自摘抄、复制、修改、发表或传播本文档内容的部分或全部，且不得以任何形式传播。

商标声明



（不完全列举）均为广州芯之联科技有限公司的商标或者注册商标。在本文档描述的产品中出现的其它商标，产品名称，和服务名称，均由其各自所有人拥有。

免责声明

您购买的产品、服务或特性应受您与广州芯之联科技有限公司（“芯之联”）之间签署的商业合同和条款的约束。本文档中描述的全部或部分产品、服务或特性可能不在您所购买或使用的范围内。使用前请认真阅读合同条款和相关说明，并严格遵循本文档的使用说明。您将自行承担任何不当使用行为（包括但不限于如超压，超频，超温使用）造成的不利后果，芯之联概不负责。

本文档作为使用指导仅供参考。由于产品版本升级或其他原因，本文档内容有可能修改，如有变更，恕不另行通知。芯之联尽全力在本文档中提供准确的信息，但并不确保内容完全没有错误，因使用本文档而发生损害（包括但不限于间接的、偶然的、特殊的损失）或发生侵犯第三方权利事件，芯之联概不负责。本文档中的所有陈述、信息和建议并不构成任何明示或暗示的保证或承诺。

本文档未以明示或暗示或其他方式授予芯之联的任何专利或知识产权。在您实施方案或使用产品的过程中，可能需要获得第三方的权利许可。请您自行向第三方权利人获取相关的许可。芯之联不承担也不代为支付任何关于获取第三方许可的许可费或版税（专利税）。芯之联不对您所使用的第三方许可技术做出任何保证、赔偿或承担其他义务。