

T/SILA

上海浦东智能照明联合会团体标准

T/SILA 023—2025

智能家居照明应用及跨生态互操作  
技术要求

Technical requirement for smart home lighting application and cross-ecosystem  
interoperability

2025-07-18 发布

2025-07-18 实施

上海浦东智能照明联合会 发布



# 目 次

|                            |     |
|----------------------------|-----|
| 前言 .....                   | II  |
| 引言 .....                   | III |
| 1 范围 .....                 | 1   |
| 2 规范性引用文件 .....            | 1   |
| 3 术语和定义 .....              | 1   |
| 4 缩略语 .....                | 3   |
| 5 Matter 跨生态互操作技术要求 .....  | 4   |
| 5.1 总体架构 .....             | 4   |
| 5.2 加密原语 .....             | 13  |
| 5.3 安全通信策略 .....           | 13  |
| 5.4 配网 .....               | 14  |
| 5.5 设备认证和操作凭证 .....        | 14  |
| 5.6 数据模型规范 .....           | 14  |
| 5.7 交互模型规范 .....           | 14  |
| 5.8 系统模型规范 .....           | 14  |
| 5.9 交互模型编码规范 .....         | 14  |
| 5.10 服务与设备管理 .....         | 15  |
| 5.11 多管理员 .....            | 15  |
| 5.12 安全要求 .....            | 15  |
| 5.13 Matter 语义标签命名空间 ..... | 15  |
| 6 照明相关应用功能集 .....          | 16  |
| 6.1 Matter 应用功能集库概述 .....  | 16  |
| 6.2 通用功能集 .....            | 16  |
| 6.3 测量与传感 .....            | 58  |
| 6.4 照明 .....               | 90  |
| 6.5 暖通空调 .....             | 115 |
| 7 照明相关设备类型库 .....          | 149 |
| 7.1 基本设备类型 .....           | 149 |
| 7.2 功用类设备类型 .....          | 151 |
| 7.3 应用类设备类型 .....          | 157 |
| 7.4 照明类设备类型 .....          | 158 |
| 7.5 智能插头/插座等执行器类设备类型 ..... | 162 |
| 7.6 开关控制类设备类型 .....        | 166 |
| 7.7 传感器类设备类型 .....         | 171 |
| 7.8 暖通空调类设备类型 .....        | 177 |
| 参考文献 .....                 | 181 |

## 前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件由上海浦东智能照明联合会提出并归口。

本文件版权归上海浦东智能照明联合会所有，未经联合会许可不得随意复制，任何单位或个人引用本标准的内容需指明标准的标准号。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件起草单位：亚数信息科技（上海）有限公司、昕诺飞（中国）投资有限公司、连接标准联盟、上海浦东智能照明联合会、OPPO广东移动通信有限公司、泰凌微电子（上海）股份有限公司、韧联数码（深圳）有限公司、深圳市华普微电子股份有限公司、深圳市朴联技术有限公司、美智光电科技股份有限公司、广州易而达科技股份有限公司、佛山电器照明股份有限公司、中山市乐式物联科技有限公司、杭州和众科技管理有限公司、祐力（中国）投资有限公司、深圳市集贤科技有限公司、横店集团得邦照明股份有限公司、非凡士智能科技（苏州）有限公司、施耐德电气（中国）有限公司。

本文件主要起草人：余宁、杨莉、宿为民、代照亮、翟新元、孟攀婷、贝华丽、茹昭、张佳婵、张凡、张喜丽、孙磊、玉鹏、冯俭智、陈曦、于金邈、吕小强、商瑞云、单晓明、李荣锋、纪瀛灏、梁裕铿、吴昊、黎禧、麦贤慧、刘杰、羊松、诸瑞雪、翁步升、吴彬强、刘卫合、张军、王昊、杨志超、魏一才、洪艳君、贾存玉、席时葭、庄晓波。

## 引 言

本文件聚焦于智能家居照明设备在应用层与传输层的互操作性需求，旨在为照明设备厂商、软件开发者和系统集成商提供统一的规范指引，推动多厂商、多平台之间的无缝对接，并保障设备在功能交互、安全认证、固件升级等关键环节的兼容性与可靠性。此外，还会注明其他引用并详细说明这些引用如何应用于本文件。

本文件已被上海浦东智能照明联合会根据与连接标准联盟（联盟）签署的谅解备忘录采纳。严禁进一步使用、修改、复制和/或分发。联盟原始的规范可以从联盟获取，网址为<http://www.csa-iot.org>。

认证需要成为联盟成员：联盟对本文件或任何联盟规范与任何设备、软件、产品或服务的认证，仅限于联盟成员，并需要遵守适用的联盟政策。

使用联盟商标和标志需要成为联盟成员：只有联盟及其成员可以使用联盟的各种商标和标志。所有联盟商标和标志必须仅按照联盟商标和标志使用指南和条款使用，该指南和条款的副本可以在[www.csa-iot.org/resources/governing-documents](http://www.csa-iot.org/resources/governing-documents)找到。

加入连接标准联盟：有关如何成为连接标准联盟成员的信息，请访问[www.csa-iot.org/become-member](http://www.csa-iot.org/become-member)。

免责声明：本文件的某些要素可能受第三方知识产权权利的约束，包括但不限于专利、版权或商标权利（该第三方可能是或可能不是联盟成员）。联盟不对识别或未能识别任何或所有此类第三方知识产权权利负责，也不应以任何方式承担责任。

联盟成员可能拥有本文件的知识产权权利，这些权利受联盟的知识产权政策约束，该政策可以在[www.csa-iot.org/resources/governing-documents](http://www.csa-iot.org/resources/governing-documents)找到。

本文件第 5 章主要定义了 Matter 跨生态互操作协议架构、安全通信机制、配网流程、数据模型等实现Matter技术的重要内容。第 6 章是针对目前Matter所支持的照明及相关设备功能集的具体定义，重点介绍了每个设备类型强制和推荐实现的功能集以及对所支持功能集的特定需求。第 7 章是针对目前Matter所支持的产品应用相关应用功能集的详细定义，重点介绍了每个应用功能集所支持的属性和命令。主要包含 照明类、插座类、开关控制类、传感器类、暖通空调类设备类型。



# 智能家居照明应用及跨生态互操作技术要求

## 1 范围

本文件规定了智能家居照明应用及跨生态互操作的核心规范、应用功能集和设备类型库。详细说明了实现应用层和传输层所需的全部内容。它旨在为实现者提供一个完整的规范，还会注明其他引用并详细说明这些引用如何应用于此规范。本文件基于 Matter 标准 1.4 版本进行编写，适用于支持无线通信协议（如 Wi-Fi、Thread等）的智能照明相关设备，尤其适用于计划接入支持 Matter 协议的智能家居生态系统的产品。

## 2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

CSA Matter 核心规范（Matter Core Specification）  
CSA 设备库规范（Device Library）  
CSA 应用功能集规范（Application Clusters）  
CSA 标准命名空间文档（Standard Namespaces）

## 3 术语和定义

下列术语和定义适用于本文件。

### 3.1

**管理员 administrator**

至少对另一个节点的access control cluster具有管理权限的节点。

### 3.2

**属性 attribute**

一种表示物理量或状态的数据实体。这些数据通过命令传递给其他节点。

### 3.3

**绑定 binding**

一个节点上的实体与另一个（或同一）节点上的一个或多个对应实体之间的持久连接。

### 3.4

**边界路由器 border router**

一种路由器，也称为边缘路由器，在两个 ip 子网之间（通常是在中心网络和外围网络之间）提供路由服务。

### 3.5

**网桥 bridge**

在fabric中代表一个或多个非matter设备的节点。

### 3.6

**认证中心 certificate authority (ca)**

颁发dac(device attestation certificate,设备认证证书)或noc(node operational certificate,节点操作证书)这样的数字证书的实体。

### 3.7

**认证声明 certification declaration**

一个数字签名的令牌，用于证明厂商认证设备的matter认证状态。

3.8

**功能集 cluster**

通过定义一个或多个属性、命令、行为和依赖关系来达成独立的功用或应用功能的规范。这个术语也可以用于端点上该规范的实现或实体。

3.9

**客户端 client**

cluster接口，通常发送命令来操作相应cluster server端的属性。一个cluster的client端与相应的具有相同cluster id的远程cluster server端通信

3.10

**服务端 server**

cluster接口，通常支持cluster的所有或大部分属性。一个cluster的server端与相应的具有相同cluster id的远程cluster client端通信。

3.11

**命令 command**

对某个值的操作请求，并预期获得带参数的响应以及带状态信息和参数的响应。

3.12

**配网节点 commissioner**

执行配网的节点角色。

3.13

**入网节点 commissionee**

正在配网以成为节点的实体。

3.14

**配网 commissioning**

分配操作节点id和节点操作凭证将节点加入fabric的一系列操作。

3.15

**配网模式 commissioningmode**

节点允许配网的模式。

3.16

**设备 device**

一个包含一个或多个节点的设备。

3.17

**鉴别码 discriminator**

12位值，用于区分多个可配网的matter设备广播。

3.18

**生态系统 ecosystem**

由一个或多个合作公司提供的一组协作的功能实体（近端或基于云），支持提供面向用户的配置、信息工具和服务。对用户来说，“生态系统”是供应商提供的“功能实体”，包括但不限于智能手机操作系统、平板、智能手机应用程序、平板应用程序、计算机应用程序、语音助手、电视界面、墙面显示/开关、云服务或跨硬件和软件平台的集成服务。

3.19

**端点 endpoint**

一个节点中可单独寻址的特定组件。

3.20

**信任域 fabric**

若干通信节点的逻辑组合，这些节点拥有共同的信任根和分布式配置状态。

3.21

**网络 network**

一组通过互联网协议具有可寻址性、连通性和可达性的节点。

## 3.22

**节点 node**

一个支持matter协议栈的可寻址实体，该实体（配网之后）有自己的操作节点id和节点操作凭证。一个设备可能承载多个节点。

## 3.23

**角色 role**

节点的一些（相关）行为集。每个节点可以具有多个角色。

## 3.24

**厂商 vendor**

生产设备的组织。

## 3.25

**厂商标识 vendor id (vid)**

一个用来唯一标识设备厂商的16位数字。

## 4 缩略语

下列缩略语适用于本文件：

ACL：访问控制列表（Access Control List）

AES：高级加密标准（Advanced Encryption Standard，来自FIPS197标准）

AP：接入点（Access Point，来自IEEE802.11-2020标准）

API：应用程序编程接口（Application Programming Interface）

BDX：批量数据交换（Bulk Data Exchange）

BLE：蓝牙低功耗（Bluetooth Low Energy）

BTP：蓝牙传输协议（Bluetooth Transport Protocol）

CA：证书颁发机构（Certificate Authority，亦称Certification Authority）

CASE：证书认证会话建立（Certificate Authenticated Session Establishment）

CAT：CASE认证标签（CASE Authenticated Tag）

CD：认证声明（Certification Declaration）

CTR：计数器模式（AES分组密码模式）（Counter Mode（AES block cipher mode，（来自NIST 800-38A标准）

DER：可辨别编码规则（Distinguished Encoding Rule，来自X.690标准）

DRBG：确定性随机比特生成器（Deterministic Random Bit Generator，来自NIST 800-90A标准）

GID：组标识符（Group Identifier，亦称“Group ID”）

GUA：全局单播地址（Global Unicast Address）

HMAC：基于密钥的哈希消息认证码（Keyed-Hash Message Authentication Code，来自FIPS198-1标准）

ICD：间歇性连接设备（Intermittently Connected Device）

ID：标识符（Identifier）

IPK：身份保护密钥（Identity Protection Key，一种跨Fabric共享的通用组密钥）

KDF：密钥派生函数（Key Derivation Function，来自RFC 5869标准）

LLA：链路本地地址（Link Local Address）

LLN：低功耗有损网络（Low power and Lossy Network）

MAC：介质访问控制（Medium Access Control）或消息认证码（Message Authentication Code）

MCSP：消息计数器同步协议（Message Counter Synchronization Protocol）

MIC：消息完整性码（Message Integrity Code，与MAC同义，用于避免与网络地址中的MAC混淆）

MRP：消息可靠性协议（Message Reliability Protocol）

NFC：近场通信（Near Field Communication）

NOC：节点操作证书（Node Operational Certificate）

OTA：空中下载（Over-the-air，多用于“空中软件更新”场景）

PAKE: 基于密码认证的密钥交换 (Password-Authenticated Key Exchange, 来自SPAKE2+标准)

PASE: 密码认证会话建立 (Passcode-Authenticated Session Establishment)

PBKDF: 基于密码的密钥派生函数 (Password-Based Key Derivation Function, 来自NIST 800-132标准)

PID: 产品标识符 (Product Identifier, 亦称Product ID)

PIN: 个人识别码 (Personal Identification Number)

PKI: 公钥基础设施 (Public Key Infrastructure)

SED: 低功耗终端设备 (Sleepy End Device)

TLV: 标签-长度-值 (Tag-Length-Value, 主要指TLV编码格式)

TRNG: 真随机数生成器 (True Random Number Generator, 来自NIST 800-90B标准)

UDP: 用户数据报协议 (User Datagram Protocol)

UGID: 通用组标识符 (Universal Group Identifier)

ULA: 唯一本地地址 (Unique local address)

UTC: 协调世界时 (Universal Time Coordinated)

VID: 厂商标识符 (Vendor Identifier, 亦称Vendor ID)

## 5 Matter 跨生态互操作技术要求

### 5.1 总体架构

#### 5.1.1 概述

Matter旨在为智能家居照明相关设备构建基于IPv6的通用通信协议。Matter协议定义了部署在设备上的应用层以及不同的链路层，用于辅助和维护不同层次的互操作性。下图1演示了常见操作的调用协议栈。

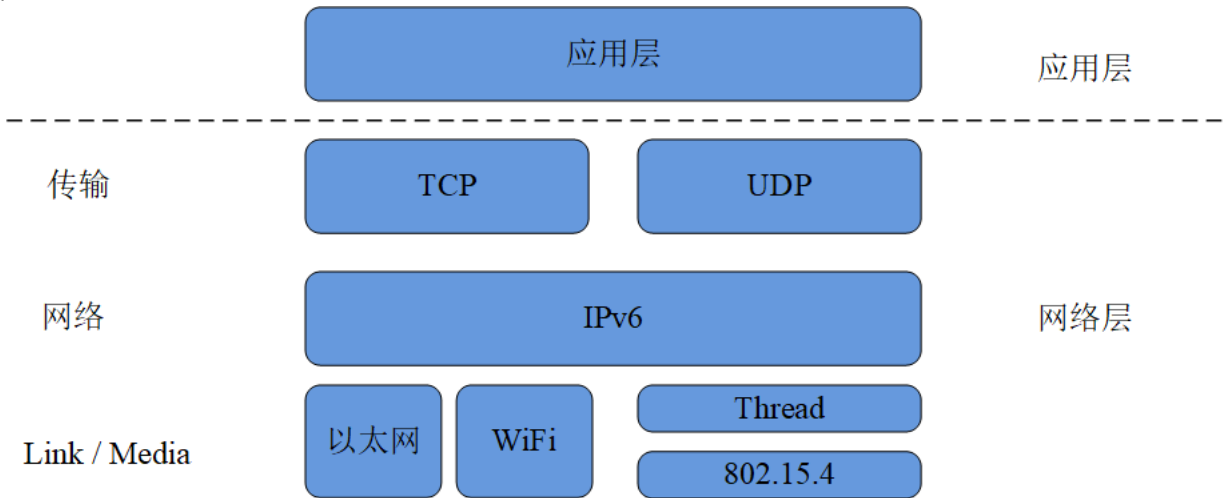


图1 调用协议栈

#### 5.1.2 分层架构

##### 5.1.2.1 概述

Matter使用分层架构分离不同的职责，并且可以在不同的协议栈之间进行了良好的封装。大部分的交互都会如图2所示。

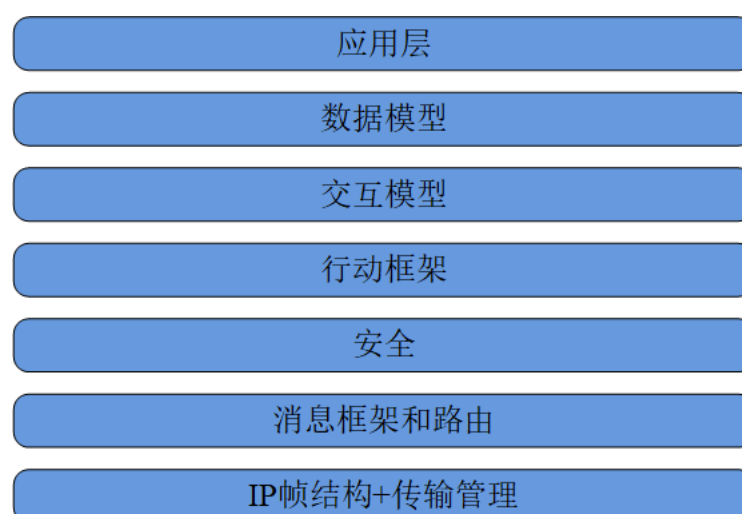


图2 分层架构

### 5.1.2.2 应用层

应用层(Application层)对应于设备的高阶业务逻辑。如照明应用的开/关灯泡及控制颜色的功能。

### 5.1.2.3 数据模型层

数据模型层(Data Model层)用来定义应用程序的数据模型及动作。应用程序将使用这些数据与设备进行互操作。

### 5.1.2.4 交互模型层

交互模型层(Interaction Model层)定义了一组可以在客户端和服务设备之间执行的交互。例如,在服务器设备上读取或写入属性对应于设备上的应用行为。这些交互作用于数据模型层定义的元素。

### 5.1.2.5 行动框架和安全层

使用交互模型构造了一个动作(action),该动作将被序列化为规定的压缩二进制格式,以便进行编码以用于网络传输。此过程在Action Framing层中处理。经过编码的action帧再由Security Layer层处理:消息被加密并附加消息身份验证码。这些处理确保消息发送方和接收方之间的数据的机密性和真实性。

### 5.1.2.6 消息框架和传输管理层

当交互被序列化、加密和签名时,Message Layer使用必需以及可选的头字段构造payload格式,其中头字段规定消息的属性以及逻辑路由信息。

当payload被Message Layer层构造后,会使用基于IP的数据传输协议(TCP协议或Matter的MRP协议)对方设备收到数据后,数据流则沿着协议栈向上移动,即各个层反转发送方对数据执行的操作,以最终将消息传递给应用程序。

除了上述捕获的数据流之外,本文件还定义了基于操作证书或密码的安全会话建立协议、组通信、适合节点之间发送大量数据的批量数据传输协议(BDX),以及用于定义制造商特定协议的条款。

## 5.1.3 网络拓扑

### 5.1.3.1 概述

#### 5.1.3.1.1 支持范围

任何支持IPv6的网络只要满足若干核心 IPv6 标准,理论上都适用于 Matter 的部署。在本文件中,重点关注三种链路层技术:以太网、Wi-Fi以及Thread。因此,本文件将主要限制在上述范围内,以便可以适当地涵盖这些链路层的配置,以适当限制认证中的测试工作量。

### 5.1.3.1.2 网络共享模型

Matter将网络视为共享资源：它没有规定专有的网络所有权或访问权限。因此，可以在同一组IP网络上覆盖多个Matter网络。

### 5.1.3.1.3 离线与受限网络的部署

Matter协议可以在没有公网IPv6基础设施的情况下运行，允许在与公网断开连接或有防火墙的网络中操作。它还可以在互联网服务提供商不支持消费者场所的IPv6或支持受限的情况下进行部署，例如，当前缀授权不能容纳本地所有的网络和设备。

### 5.1.3.1.4 跨子网本地通信

Matter协议支持跨一个或多个IPv6子网的本地通信。典型的网络可以支持有多个子网组成的交叉Fabric网络，可能包括Wi-Fi/以太网子网，或者一个或多个低功耗有损网络（LLN）子网等。在本文件中，Thread是受支持的LLN标准。

### 5.1.3.2 单一网络拓扑

在单一网络拓扑中，所有的Matter设备都连接到一个单一的逻辑网络。它可以是Thread/802.15.4网络、Wi-Fi网络或以太网网络。在Wi-Fi/以太网网络的情况下，网络实际上可以跨越多个Wi-Fi和/或以太网段，前提是所有段都在链路层桥接。节点（Node）是Fabric中的Matter设备的单个实例，可在IP网络上运行。

单一网络拓扑中的每个节点都通过单个网络接口与Fabric中的每个其他节点进行通信。

单一Thread网络拓扑如图3所示，单一Wi-Fi网络拓扑如图4所示。

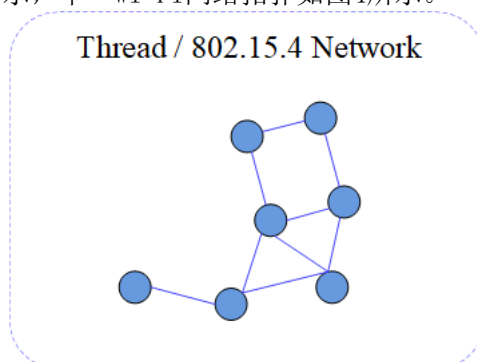


图3 单一 Thread 网络拓扑

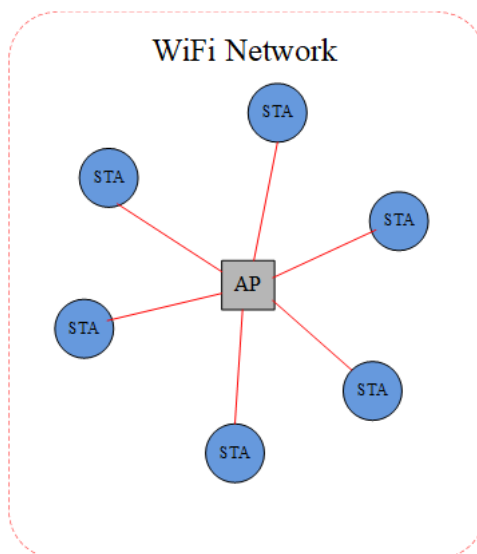


图4 单一 Wi-Fi 网络拓扑

### 5.1.3.3 星形网络拓扑

#### 5.1.3.3.1 概述

星形网络拓扑（如图5）由多个外围网络组成，这些网络通过Hub连接在一起。Hub通常是客户家庭网络（Wi-Fi/以太网）中的设备，而外围网络可以是任何支持的网络类型。外围网络必须始终通过一个或多个边界路由器（Border Router）直接连接到Hub。

#### 5.1.3.3.2 网络架构

在架构上，任一数量的外围网络可以存在于单个Fabric中，包括相同类型的多个网络。节点可以具有到任何网络（Hub或外围设备）的接口，并且可以直接与同一网络上的其他节点通信。然而，任何必须跨越网络边界才能到达目的地的通信必须通过边界路由器（Border Router）完成。

#### 5.1.3.3.3 协议要求

该协议对边界路由器提出了一系列要求。这些要求涉及地址分配、路由分配和通告、多播支持和代理发现。

注：在本文件中，Thread是主要支持的LLN。在多数情况下，客户安装将尝试维护一个简单的网络拓扑，包括一个单Wi-Fi/以太网子网和一个单Thread网络。但是，可以支持多个Thread网络。

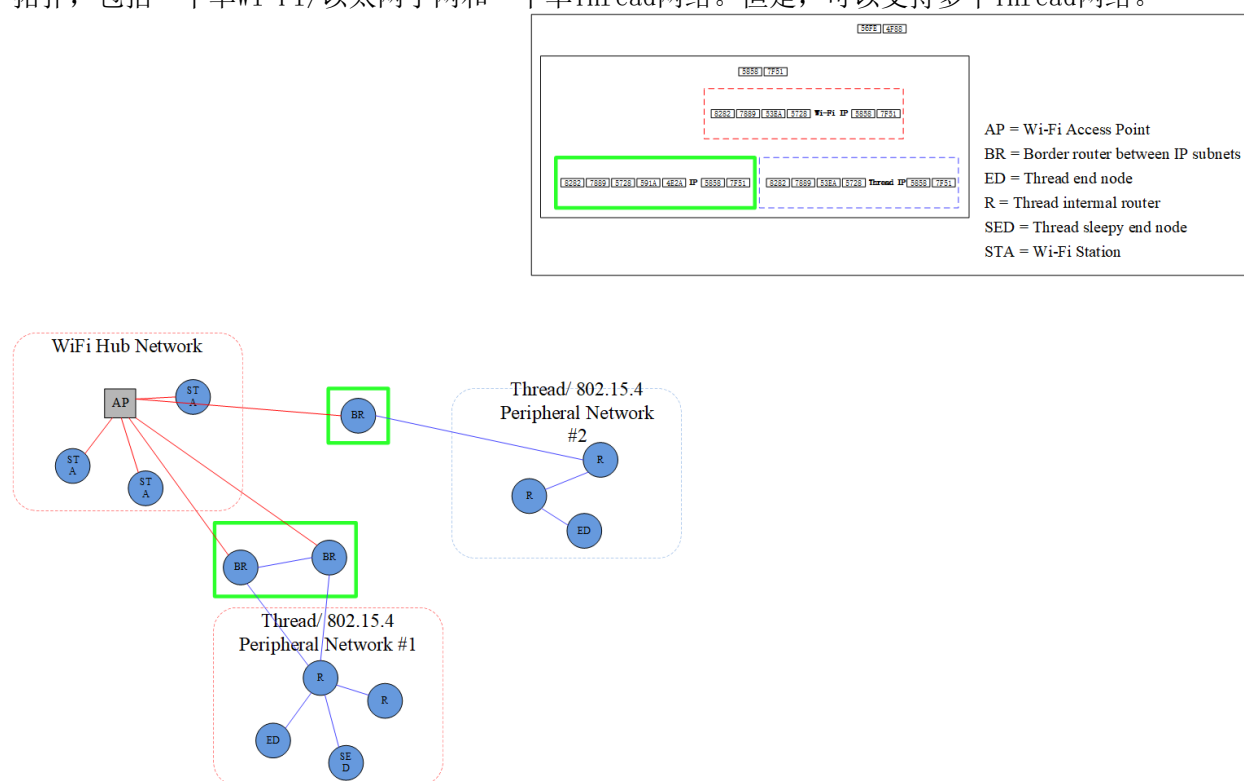


图5 星形网络拓扑图

### 5.1.4 命名域

#### 5.1.4.1 概述

Matter协议架构中明确包含多管理员支持机制（multiple administrators），需注意的是，不是任何公共信任根（multi-admin）。多管理员通过multiple fabrics实现，建立在命名域、Key、onboarding、CASE通讯安全、数据模型（如fabric范围数据）几个方面的基础上。

#### 5.1.4.2 通过 Fabric 构建信任

Fabric是共享同一个Trusted Root的Matter设备的集合。Matter中Trusted Root作为根CA，颁发节点操作证书（NOC），识别节点身份。在一个Fabric内，每个节点都有一个唯一标识Node ID。Fabric作为一个命名空间来管理所有权，在Fabric范围内使用标识符确保资源的分配和选择的唯一性。

根证书颁发机构（CA）颁发自己的根公钥（Root PK）用以表达Fabric范围的唯一标识，所以CA也会被称为操作的信任根。所有公钥基础设施（PKI）的安全性取决于受保护的CA的私钥，既不可猜测也不可获得；该属性反过来意味着公钥是全局唯一的。在任何根CA中，Fabric是由64-bit数字唯一标识的。Commissioner与Commissioner的协作采用一套基于唯一根CA下的机制，Matter协调Commissioner关联的根CA以及其他关联的数据存储称之为“可管理域的管理”（ADM）。只要所有标识符的分配复合Fabric唯一性，Matter规范并不要求ADM的算法细节和策略。Fabric中唯一的根CA关联一个唯一的Fabric ID，在每个Fabric中ADM会给每个节点分配一个Operational Node ID。

此处概述的作用域策略确保每个作用域标识符可以仅由负责作用域的实体分配，而不考虑冲突或伪造。例如，两个不同的CA可以分配相同的Fabric标识符，这不会给网络内的设备造成任何问题。责任的范围授权还为删除特定标识符提供了明确的指导方针。

### 5.1.4.3 作用域策略

Matter设备可以是多个Fabric的成员，因此具有多个关联的Node ID。作用域策略还自然有助于明确解析名称和凭据，并在每个Fabric的相关ADM服务上确定名称空间的责任。

### 5.1.4.4 初始 Fabric ID 配网

在第一次配网之前，例如在出厂重置状态下，一个典型设备不包含预先分配的信任操作根，也不包含Fabric ID和Node ID形式的操作标识。然而，各种交互都需要Fabric ID或Node ID。这些标识符出现在许多内部Fabric中——从地址发现到识别安全会话，再到评估访问控制权限。为了规范与设备的所有交互并解决bootstrap问题，保留了一个特殊的原始Fabric ID，并将一组初始访问控制权限与配网会话相关联。

### 5.1.5 唯一标识

#### 5.1.5.1 Fabric 引用和 Fabric 标识符

Fabric ID是在根CA范围内唯一标识Fabric的一个64-bit的数字。从概念上来说，Fabric引用包含根证书颁发机构的公钥和Fabric ID。因为定义了许多用于压缩参考的机制，所以Fabric引用使用起来很麻烦。压缩形式的Fabric引用用于隔离操作发现期间不相关的设备集合（一种名称空间形式）。

数字0是Matter保留的Fabric ID，不能在产品中使用。

在数据模型中Fabric包括节点的集合，节点之间可以基于数据模型的元素进行交互。

在数据模型的下层，将数据模型的交互作为消息传递，层层传递Fabric相关或不相关的消息。

示例：数据模型的消息Fabric ID为0时表示未指定Fabric。

#### 5.1.5.2 Vendor 标识符

Vendor标识符（Vendor ID或VID）是16-bit的数字用于唯一标识一个具体的厂商、组。其中Vendor ID是CSA联盟颁发的（参见“2.1 规范性引用文件”）。

下表1的Vendor ID是Matter保留的，其他所有Vendor ID都被定义在“2规范性引用文件”。

表1 Vendor ID 分配

| 范围             | 类型                    |
|----------------|-----------------------|
| 0x0000         | Matter标准              |
| 0x0001-0xFFFF0 | 依据CSA给厂商颁发的编码保留给厂商的代码 |
| 0xFFFF1        | 用于测试的Vendor#1         |
| 0xFFFF2        | 用于测试的Vendor#2         |
| 0xFFFF3        | 用于测试的Vendor#3         |
| 0xFFFF4        | 用于测试的Vendor#4         |

注：这里说的用于测试的Vendor IDs是Matter保留的给设备厂商和业余爱好者用于开发和测试的。

### 5.1.5.3 产品标识符

产品标识符（Product ID或PID）是16-bit的数字用于标识一个Vendor下的产品。Product ID是Vendor分配管理的，在Vendor范围下每个产品必须唯一。Product ID在同一个Vendor ID下不能重复使用于不同的产品。Product ID是描述设备的类型所以可以不与物理设备严格对应，即一个Product ID可以对应多个物理设备。

Product ID不能使用0x0000，因为0x0000被用在如下几个场景：

- 在设备发现时用来表达一个匿名的Product ID；
- OTA是用来表达是多个Product ID；
- 避免多节点Enhanced Commissioning Method (ECM)的Onboarding Payload的混乱。

### 5.1.5.4 Group 标识符

#### 5.1.5.4.1 概述

Group标识符（Group ID或GID）是16-bit的数字用于在消息层标识Fabric范围内节点的集合。Group ID未来还可以在交互层绑定每个节点下一个或多个Endpoint。

Group ID的分配控制如下表2所述。

表2 Group ID 的分配

| 范围            | 类型                                    |
|---------------|---------------------------------------|
| 0xFF00-0xFFFF | 保留用于静态组播（staticmulticast）与任播（anycast） |
| 0x0001-0xFEFF | 在应用程序中由FabricAdministrator分配的Group ID |
| 0x0000        | 空或者未定义的Group ID                       |

#### 5.1.5.4.2 通用 Group ID

通用Group ID（UGID）是本文件中保留的16-bit子范围，在Fabric中所有节点都知道这些特殊的多播、组播、任播。通用Group ID的分配在表3中描述。

表3 通用 Group ID 分配

| 范围            | 类型                 |
|---------------|--------------------|
| 0xFFFF        | AllNodes           |
| 0xFFFE        | Allnon-sleepyNodes |
| 0xFFFD        | AllProxies         |
| 0xFF00-0xFFFC | 保留，未来使用            |

Commissioner应该在Fabric内在所有节点为这些Group配置一个或多个共享密钥，由于密钥和IPv6组播前缀在不同结构之间是不同的，因此通用组只允许在特定Fabric内的通信。

#### 5.1.5.4.3 所有节点的 Group

此Group用于在Fabric内所有节点上收发消息。

#### 5.1.5.4.4 所有非睡眠节点 Group

此Group用于在Fabric内所有支持电源的节点上收发消息。可休眠的节点不得订阅此组。

#### 5.1.5.4.5 所有代理 Group

此Group用于在代理订阅中发现代理节点。

### 5.1.5.5 Node 标识符

#### 5.1.5.5.1 概述

Node标识符（Node ID）是一个64-bit的数字用来标识Fabric内或组内的一个节点。表4展示了Node标识符的分配。

表4 Node 标识符分配

| 范围                                             | 类型                     |
|------------------------------------------------|------------------------|
| 0xFFFF_FFFF_FFFF_xxxx                          | 组Node ID               |
| 0xFFFF_FFFF_0000_0000_to_0xFFFF_FFFF_FFEF_FFFF | 保留，未来使用                |
| 0xFFFF_FFFE_xxxx_xxxx                          | 本地临时Node ID            |
| 0xFFFF_FFFD_xxxx_xxxx                          | CASE Authenticated Tag |
| 0xFFFF_FFFC_xxxx_xxxx_to_0xFFFF_FFFC_FFFF_FFFF | 保留，未来使用                |
| 0xFFFF_FFFB_xxxx_xxxx                          | PAKE key标识             |
| 0xFFFF_FFF0_0000_0000_to_0xFFFF_FFFA_FFFF_FFFF | 保留，未来使用                |
| 0x0000_0000_0000_0001_to_0xFFFF_FFEF_FFFF_FFFF | Operational Node ID    |
| 0x0000_0000_0000_0000                          | 未定义的Node ID            |

其中，

- Node ID用于在内部的API、在数据模型的核心字段，并且用于解析操作的IPv6地址。Node ID的取值范围从 0xFFFF\_FFF0\_0000\_0000 到 0xFFFF\_FFFF\_FFFF\_FFFF，并且 0x0000\_0000\_0000\_0000是保留的值。
- Operational Node ID是64-bit的数字标识Fabric内的唯一节点。所有的消息必须包含一个Operational Node ID作为源地址。所有的单播消息必须包含一个Operational Node ID作为目的地址。当来源或者目的地址被省略时，必须可以没有歧义的通过Session ID推导出来。
- Group Node ID是一个64-bit的Node ID，在Node ID的低半段地址位中包含了Group ID。
- Temporary Local Node ID是一个64-bit的Node ID，在后32位中包含一个与实现相关的值。这使得系统在Operational Node ID不可用的情况下，仍可用于追踪连接、传输层链路等内部管理用途。
- PAKEkey标识，Node ID的一个子集，用于将访问控制对象分配给特定的PAKE key。如在配网时需要在Comissionee给任意一个Comissioner创建一个administrative的ACL权限，用于通过PIN码来建立PASESession。

#### 5.1.5.5.2 CASEAuthenticatedTag

Node ID的一个子集，用于将访问控制对象分配给给一个Group，这个Group内的每个Node共享一个CASE Session。

#### 5.1.5.5.3 未定义的 Node ID

未定义的Node ID (0x0000\_0000\_0000\_0000) 是一个保留的值，永远不会出现在消息里，协议也不会使用。它只是表示为初始化、丢失、或者一个无效的Node ID。

#### 5.1.5.6 IPv6 寻址

##### 5.1.5.6.1 概述

本协议使用IPv6地址进行操作通信。Node ID和Fabric ID被解析为各种类型的IPv6地址RFC4291。

##### 5.1.5.6.2 IPv6 单播地址

IPv6单播地址是一个在IPv6网络上唯一识别一个单节点的地址。本文件一个主要的设计目标是允许节点利用本机的IPv6技术。因此，应使用在节点之间提供连接和可路由性的操作IPv6单播地址。这包括全局单播地址（GUA）、链路本地地址（LLA）或唯一本地地址（ULA）。

##### 5.1.5.6.3 IPv6 组播地址

IPv6组播地址是使用基于单播前缀的IPv6组播地址[RFC 3306]形成的：

- 前12bit由RFC3306定义，且为0xFF3
- 接下来的4bit是“scop”（作用域），并基于RFC7346第2节设置为：Site-Local (0x5)-跨越结构中的所有网络，包括Thread、以太网和Wi-Fi
- 接下来的8bit将被保留 (0x00)。
- 接下来的8bit被全权管理，设置为0x40标识一个64-bit的网络前缀

多播地址的网络前缀部分是通过连接形成的64-bit位字符串：

- 0xFD用于本地制定ULA前缀，详情参见“RFC4193的3.1节”）
- 采用大端序且高56位是Fabric ID

多播地址的32-bitGroup标识符是由以下部分组成的：

- 低8位是Fabric ID
- 0x00
- 剩余的16位是大端序的Group标识符，参见“5.1.5.4 Group标识符”。

示例：在给定<Fabric ID>和<Group ID>时，是本地域的多播地址：FF35: 0040: FD<Fabric ID>00: <Group ID>。

注：虽然总是使用Site-Local域，但是通过设置IPv6跳计数可以限制有效范围。

多播地址的格式确保降低一个节点接收到它不感兴趣的多播消息的概率。如果在多播地址上确实发生了冲突，会根据消息的64-bitMIC来检查对应的操作组密钥并消除歧义。

#### 5.1.5.6.4 IPv6 多播端口号

IANA给Matter分配的端口号是5540。

#### 5.1.5.6.5 IPv4 共存

Matter设备应能容忍IPv4地址，并且可以忽略这些地址。

### 5.1.6 设备标识

每个Matter设备都包含多个证书链。

设备认证证书（DAC）用来验证制造商的真实性以及设备硬件和软件的认证状态。设备认证证书用于Commissioner在配网时确保可信的设备能加入Fabric。

每个Matter设备都会获得一个Operational Node ID和一个NOC。NOC在Fabric下通过加密方式将唯一的运行时操作密钥与Operation Node ID进行绑定来标识自己，并可以通过受信任的证书颁发机构（CA）的签名进行验证。Operational Node ID会在设备恢复出厂设置或移除Fabric时被删除。在设备配网进入Fabric过程中发出NOC。这些步骤有助于保护最终用户的隐私，并适应不同的信任模型。

### 5.1.7 安全

Matter使用现代的安全策略来保护Fabric。Matter指定了“5.2 加密原语”章节中详细介绍的安全原语以提供全面的保护。公钥协商和数字签名基于NISTP-256曲线（secp256r1）的椭圆曲线非对称加密算法。操作中数据传输使用常用的AES对称加密算法。在需要密码增强的授权场景中，Matter使用一种有效的增强PAKE算法SPAKE2+。

核心的密码学原语构成了Matter互相补充的安全协议基础。所有的单播消息都经过安全保护、身份验证并提供重放保护。在IPv6多播情况下，Matter提供了组传递机制，对于在LLN上有效寻址非常有用。组消息传递有限考虑数据包处理的低延迟。

### 5.1.8 设备配网

设备配网（参见“5.4 配网”）是让一个设备加入Fabric的过程。被配网的设备称为Commissionee，配网的设备称为Commissioner。设备配网包括以下步骤：

- 1) 设备发现：Commissioner 在网络上发现 Commissionee，如低功耗蓝牙、Wi-Fi 或已经连接的IP 网络。Commissioner 从 Commissionee 设备上通过二维码、手工配对码、NFC 或者其他方式获得 Passcode。Passcode 用于通过 Passcode-Authenticated Session Establishment（PASE）来建立一个安全的 CommissioningSession。发现设备与得到设备的 Passcode 的顺序并不重要；
- 2) 使用 PASE 建立安全通道：使用 PASE 在 Commissioner 与 Commissionee 之间建立加密密钥。Commissioner 与 Commissionee 之间的所有消息将会基于 PASE 密钥进行加密。这个过程中会进行设备认证；
- 3) 设备认证：Commissioner 确认 Commissionee 是一个可信的认证设备，如果未认证则通知用户；
- 4) 信息配置：Commissioner 向 Commissionee 提供监管域、UTC 时间、操作证书和网络接口配置等信息；

- 5) 加入网络:Commissioner 通知还未入网的 Commissionee 去连接操作的网络。此时 Commissioner 或管理员就可以通过 IPv6 地址去使用（如果已知）或发现（如果未知）Commissionee;
- 6) 建立 CASE 安全连接: 增强的密钥用于在 Commissioner 与 Commissionee 之间或 Administrator 与节点之间建立 CASE 安全通讯。所有的 Commissioner 或 Administrator 与节点之间的单播信息都是基于 CASE 的密钥;
- 7) 配网完成消息交换: 在操作网络上使基于 CASE 的加密密钥加密的消息交换, 意味着配网过程成功完成。

在配网完成后或在建立了加密密钥的CASE的通道上,Commissioner可以在操作网络对Commissionee进行多次配网。

### 5.1.9 间歇性连接设备

该规范的一个目标是为与间歇性连接到网络或在一段时间内无法访问的设备进行通信提供支持。间歇性连接设备（ICD）操作模式被定义为帮助优化此类节点的网络流量、可达性和电源消耗。间歇性连接可以由许多因素引起, 包括通过循环工作来最小化功耗、网络连接的间歇性可用性、设备的移动性或电源的间歇性可用性。

ICD操作模式被设计为与底层网络层机制无关, 可以在任何支持的IP接口（包括Thread和Wi-Fi）上使用。

注: 由于客户端和基础设备可能有限的缓冲空间来缓存间歇性连接设备的信息, ICD通信模式应设计为ICD主要作为发起者。

——可休眠终端设备（SED）

SED工作模式由Thread标准定义, 旨在帮助Thread设备在电池或有限能量收集等受限电源上运行时, 延长并优化电池寿命。虽然Matter ICD工作模式可以利用Thread睡眠端设备（SED）行为来支持Thread ICD设备, 但不宜将二者混淆。

### 5.1.10 根数据模型

Endpoint0 (zero) 应为Root Node end point。

Endpoint0 (zero) 应支持Root Node device type。

### 5.1.11 栈约束

#### 5.1.11.1 系统模型约束

##### 5.1.11.1.1 访问约束

节点在每个Fabric下应该至少生成4个访问控制条目。

示例: 一个支持 5 个 Fabric 的节点必须支持至少 20 个 ACL 条目, 如果它支持 N 个条目, 则必须强制任意 K 个 Fabric 一起使用不超过  $N-4*(5-K)$  个 ACL 条目。

具体的设备类型可能会对ACL记录有额外的要求。

##### 5.1.11.1.2 Group 约束

节点在每个Fabric应该支持一个Group Key用于管理IPK。

如果一个节点实现的一个或多个设备类型支持Groups cluster, 节点应该再额外支持设备类型所要求Group的最大数量, 且以下几点不能低于强制最低要求:

- 节点应在每个Fabric中支持至少3个Group Key。
- 节点应在每个Fabric的每个具有Groups cluster实例的端点上支持至少4个Group。

一个支持5个Fabric并且有1个具有Groups cluster的端点的节点, 必须支持至少20个组表项。而一个支持5个Fabric并且有2个具有Groups cluster的端点的节点, 则必须支持至少40个组表项。

- 节点应该支持IPv6多播组员, 如果每个操作Group都支持的话。
- 支持Group Key Multicast Policy的临时字段Group Key Set Struct。

### 5.1.11.2 交互模型约束

#### 5.1.11.2.1 读交互约束

Server应确保每个Fabric的节点再配网时能持有一个读交互客户端,这个客户端可以从一个Fabric获取最多9个路径。

Server可允许读交互即使在没有可访问Fabric的情况下,也可使用某些资源(例如,使用PASE时)。

### 5.1.11.2.2 订阅交互约束

订阅交互应遵循以下约束:

- 发布者应确保每个Fabric下配网的节点在至少支持3个订阅交互到发布者,每个订阅应至少支持3个属性/事件路径。
- 即使在没有可访问的Fabric、可用的资源(如PASE)时,Server也应该允许订阅交互。
- 设备类型规范可能需要一个更大的数字支持订阅或者路径。
- SUBSCRIPTION\_MAX\_INTERVAL\_PUBLISHER\_LIMIT定义发布者为任何订阅选择的最大间隔的上限。这应设置为60分钟。
- 受上述最小约束的最小支持能力,在基本信息Cluster的Capability Minima属性中报告。

### 5.1.11.2.3 调用交互约束

调用请求Action应该被限制为单个的具体命令路径。

## 5.2 加密原语

本节介绍Matter协议中使用的各种加密原语、算法和协议构建块,并为每个加密原语、算法和协议构建块引入一个功能抽象,可在本文件的其他章节中引用。本章还将这些加密原语映射到具有相应信息性或规范性引用的特定实例。在相关的地方,它还提供了关于在特定上下文中使用这些映射以实现兼容实现的必要或相关信息。

给定消息格式的版本,加密原语被映射到特定的实例。本协议中没有加密套件协商:消息格式的1.0版有一个本章定义的加密套件。

每个小节都以通用方式定义了密码原语,并将其具体映射到消息格式的1.0版的特定实例上。本章还可作为设备需要支持哪些加密原语的指南,但必须注意,并非所有设备都必须支持全部原语。例如,设备可能不需要Crypto\_PBKDF()原语,因为在某些情况下,基于该操作的值可以在设备的制造过程中预先计算和存储。本章中提出的函数映射对于由函数计算的值是规范的,但对于函数在实现中的接口方式是信息性的。例如,一个函数如果操作成功既可以返回一个boolean也可以返回一个值表示成功,还可以使用异常机制来代替返回boolean。

还有必须要注意的是,并非所有的密码原语都对规范的其他部分可见。例如,不得在Crypto\_DRBG()实现之外调用Crypto\_TRNG()原语。

加密原语对主机本地的数据进行操作。如果存在更复杂的数据类型,并且适用于它们的外部表示,本章将说明编码的细节。没有任何附加上下文的简单多字节数据类型在过程内部使用时,假定是主机的字节顺序,除非另有说明。

所有的八进制字符串都是从索引0开始的,并且对于长度为N的八进制字符串,按从左到右的顺序表示索引0到N-1的值。

## 5.3 安全通信策略

安全信道和消息层提供了一个一致的网络服务基板,以允许节点之间进行安全通信。

在配网和单播通信期间,提供了一个发现机制来确定对等节点的IPv6地址和操作参数,使用基于共享密码(PASE)或证书(CASE)的安全会话建立机制。

本章主要介绍消息、安全信道协议和消息通信三部分内容。

- 消息部分主要介绍消息的帧格式、消息计数器、消息处理、消息安全、消息隐私以及消息交换。
- 安全信道协议部分主要介绍五个重要的安全信道协议,包括:
  - 消息计数器同步协议(MCSP)
  - 消息可靠协议(MRP)
  - 基于密码验证的会话建立(PASE)
  - 基于证书认证的会话建立(CASE)
  - 蓝牙传输协议(BTP)

- 消息通信部分主要介绍单播通信、组播通信以及组密钥管理。

#### 5.4 配网

本节的目的是定义配网信息的内容，这些配网信息用于将设备接入Matter网络。它还指定了配网信息的表示方式和编码方式，包括二维码、手动输入的代码以及NFC标签中的内容。

配网信息由配网节点使用的必需和可选信息组成，是用来确保commissoner和commissioneer之间的交互，并提供一致的用户体验。这些内容的一部分或全部会被编码成不同格式，一些人类可读的格式（如数字字符串），一些机器可读的格式（如二维码和NFC），用于在设备上打印或显示，或是集成到设备中。以下是可在Matter设备的配网信息中使用的元素。

在配网信息里可能包含以下内容：版本（Version）、厂商标识（Vendor ID）、产品标识（Product ID）、客制化流程（Custom Flow）、发现功能位掩码（Discovery Capabilities Bitmask）、鉴别码（Discriminator value）、密码（Passcode）、TLV数据（TLV data）。

#### 5.5 设备认证和操作凭证

本节描述了在实体之间建立信任所涉及的程序和加密凭证。

设备认证部分为配网节点和管理员（Administrator）提供了在共享密钥和其他凭证等敏感信息之前确定节点是否是真实的Matter认证产品的机制。设备认证特性依赖于DAC链和认证声明（CD，Certification Declaration）。

节点操作凭证部分描述了所有节点在证书认证会话建立期间相互认证所使用的凭证，包括NOC链。这些凭证构成了节点如何识别和参与安全操作单播通信的基础。

#### 5.6 数据模型规范

本章是数据模型规范的一部分，这些规范与底层（编码、消息、网络、传输等）无关。以下每条规范内容都可以独立维护。这部分内容，作为一个整体，应被独立维护，与下层无关且解耦。这部分内容可被一组协议栈规范所包含和引用，以得到一个完整的垂直标准。

数据模型：为端点、cluster、属性、数据类型等定义了第一级元素和命名空间。

交互模型：定义了节点之间的各种交互，会话和行动。

系统模型：定义endpoint和clusters之间的管理关系。

应用功能集：cluster规范的参考库。

设备类型库：设备类型规范的参考库。

#### 5.7 交互模型规范

本节是数据模型规范很重要的一部分，但是不涉及到底层（编码、消息、网络、传输等）的内容。以下列表中的每个规范是可以独立维护的。而这几个独立的规范又可以作为一个整体，而不应该受到任何来自底层的影响来单独进行维护。这个整体有时候会被不同的底层协议栈进行适配，比如数据模型规范同时适用于Zigbee和Matter。

#### 5.8 系统模型规范

本节是数据模型规范的一部分，这些规范与底层（编码、消息、网络、传输等）无关。以下每条规范内容都可以独立维护。这部分内容，作为一个整体，应被独立维护，与下层无关且解耦。这部分内容可被一组协议栈规范中包含的方式所引用，以得到一个完整的垂直标准。

数据模型：为端点、功能集、属性、数据类型等定义了一阶元素和命名空间。

交互模型：定义了节点之间的交互、会话和行动。

系统模型：定义了端点和功能集之间的管理关系。

功能集库：功能集规范的参考库

设备库：设备类型定义的参考库

#### 5.9 交互模型编码规范

本节详细定义了Matter TLV格式中交互模型的编码规则。

具体来说，将会详细定义Matter交互模型中使用的消息帧中的应用数据帧的编码格式。消息帧中帧头的详细定义在消息帧格式中介绍，而不在此处提及。

## 5.10 服务与设备管理

本节是对服务和设备管理有关功能集的详细定义，比如Basic Information Cluster用来提供如Vendor ID、Product ID、序列号、软硬件版本等这样的节点基本信息，Group Key Management Cluster用来管理节点的组密钥，Thread/Wi-Fi/Ethernet Network Diagnostics 功能集用来帮助用户或管理员诊断潜在的Thread/Wi-Fi/Ethernet网络问题，Administrator Commissioning 功能集用来触发节点使其能够被新的管理员配网

## 5.11 多管理员

多管理员（Multi Fabrics）特性允许一个节点被多个独立管理的Fabric配置（commissioned）。使用此特性，当前管理员可以（在用户同意的情况下）允许另外一个fabric的管理员对其fabric上的节点进行配网。新的配网节点必须拥有其受信任的根证书颁发机构（TRCA）颁发的自己的NOC。commissioning完成且节点被正确配置，新加入的Fabric上的管理员可以访问节点并执行所有管理任务。

Fabric由其受信任的根证书颁发机构（TRCA）锚定。TRCA可以委托一个或多个中间证书颁发机构（ICA）颁发节点互操作证书集（NOCs）。多个供应商或公司可以使用相同的CA层次结构，在这种情况下，它们将受相同的受信任根证书颁发机构管理。

## 5.12 安全要求

每个Matter安全和隐私要求都引用了在威胁模型中促使需要该要求的底层对策（CM）和威胁（T）。要求按主题分组。除非另有说明，否则这些要求通常适用于所有设备和节点（即所有Matter功能的实现）。某些要求会专门针对特定的实现组或角色进行特别说明。

## 5.13 Matter 语义标签命名空间

本章介绍了语义标签功能的一部分——命名空间，详见表5，可参见“2.2 资料性引用文件”。

标准命名空间，它们包括通用命名空间和特定设备命名空间。

通用命名空间以“NamespaceID0x01”开头，包含适用于任何领域的语义标签。例如包括像“左”、“右”、“上”和“下”的方向词，或者像“内部”和“外部”的位置词。

特定设备命名空间以“NamespaceID0x41”开头。在分类命名空间中定义的语义标签应仅限于在各种设备类型或设备类型集内部使用。

注：一些特定于某些设备类型组（与能源和洗衣相关）的命名空间已从公共范围分配了一个ID，即使它们仅适用于一组特定的设备类型。

一个标签列表可以将多个标签组合在一起，根据设备的适用情况进行合适的组合。对于任何给定的设备类型，标签来自于该设备类型的命名空间，以及任何通用命名空间和/或制造商特定的命名空间。例如，一个有两个照明装置单元的室外照明灯具，其中一个向上照射，另一个向下照射。其中一个照明单元将会以一个具有标签列表的端点来表示，其标签列表具备带有标签“Location.Outdoor”、“Position.Top”和“Direction.Upward”的标签结构（Tag Structs）；而另一个照明单元也会以一个具有标签列表的端点来表示，其标签列表具备带有标签“Location.Outdoor”、“Position.Bottom”和“Direction.Downward”的标签结构。

表5 matter 语义标签命名

| ID       | 命名空间       | 概述                      |
|----------|------------|-------------------------|
| 通用命名空间   |            |                         |
| 0x01     | 通用开合命名空间   | 用于描述与关闭和开启相关的事物的标签。     |
| 0x02     | 通用罗盘方向命名空间 | 用于描述与罗盘方向相关的事物的标签。      |
| 0x03     | 通用罗盘位置命名空间 | 用于描述与罗盘位置相关的事物的标签。      |
| 0x04     | 通用方向命名空间   | 用于描述与方向相关的事物的标签。        |
| 0x05     | 通用等级命名空间   | 用于描述与等级相关的事物的标签。        |
| 0x06     | 通用位置命名空间   | 用于描述与位置相关的事物的标签。        |
| 0x07     | 通用数字命名空间   | 用于描述与编号相关的事物的标签。        |
| 0x08     | 通用定位命名空间   | 用于描述与定位相关的事物的标签。        |
| 0x0A     | 电气测量命名空间   | 用于描述电力负荷的标签。            |
| 0x0E     | 洗衣设备命名空间   | 用于描述洗衣设备类型的标签。          |
| 0x0F     | 电源设备命名空间   | 用于描述电源设备类型的标签。          |
| 0x10     | 通用区域命名空间   | 用于描述与家庭区域相关的标签。         |
| 0x11     | 通用地标命名空间   | 用于描述与家庭地标相关的标签。         |
| 0x12     | 通用相对位置命名空间 | 用于描述与设备外部参照物相对位置的相关的标签。 |
| 特定设备命名空间 |            |                         |
| 0x41     | 冰箱命名空间     | 用于描述制冷设备类型的标签。          |
| 0x42     | 室内空调命名空间   | 用于描述室内空调设备类型的标签。        |
| 0x43     | 开关命名空间     | 用于描述开关设备类型的标签。          |

6 照明相关应用功能集

6.1 Matter 应用功能集库概述

本文件规定了Matter应用功能集库（MACL）。MACL是由连接标准联盟开发的功能集库，是一个工作库，随着新功能的添加而定期更新。开发人员在构建新应用时应使用MACL来查找可以纳入新应用的相关功能集。相应地，为应用定义的新功能集也应考虑纳入MACL中。MACL由多个功能集组合而成。在许多应用领域普遍有用的功能集被包含在通用集合中。主要用于特定应用领域的功能集则被分组到面向领域的集合中。

6.2 通用功能集

6.2.1 概述

功能集库是由各个功能集的单个章节组成的。通过功能集库中的文档，以获取所有章节列表的文件信息。章节之间的关系是使用X.Y表示法进行的，其中X是章节，Y是该章中的小节。对外部文档的引用参见“2 引用文件”中，并使用[Rn]符号。

6.2.2 通用描述

6.2.2.1 概述

本文档中的功能集是一些具有足够通用性的接口，以供大部分的产品应用来使用。

6.2.2.2 功能集列表

本章中规定的通用功能集如表6所示。

表6 通用功能集概览

| 标识符    | 功能集名称                       | 描述                                         |
|--------|-----------------------------|--------------------------------------------|
| 0x0003 | Identify                    | 包含使设备处于识别模式的属性和命令（比如，灯语）                   |
| 0x0004 | Groups                      | 管理相关联的节点加入一个或多个分组，并可以进行组播交互                |
| 0x0006 | On/Off                      | 管理开关设备的开关状态                                |
| 0x0008 | Level Control               | 如果设备具有一些可以离散变化的特性值，那么这个功能集可以规定范围之内设置这个特性的值 |
| 0x0045 | Boolean State               | 定义布尔态变量的属性和事件                              |
| 0x0080 | Boolean State Configuration | 配置相关布尔态变量的属性和命令                            |
| 0x003B | Switch                      | 各种类型的开关设备的属性和事件                            |
| n/a    | Alarm Base                  | 基本警报功能集，从中演化出所有警报功能集                       |

## 6.2.3 标识功能集

### 6.2.3.1 概述

这个功能集支持一个端点标识状态（如，闪烁灯光），向观察者（如，安装人员）指示它是多个节点和/或端点中的哪一个。它还支持一个多播请求，要求任何正在标识自己的端点响应发起者。

这个功能集的状态可以在一个节点上的多个端点上共享。

**示例：**一个节点上有两个端点，一个是温度传感器，另一个是湿度传感器，它们可以共享相同的功能集实例，因此也可以共享标识状态（例如，节点上的单个LED灯）。

### 6.2.3.2 分类

标识功能集分类信息详见表7。

表7 标识功能集分类表

| 等级   | 角色      | 范围       | PICS 编码 |
|------|---------|----------|---------|
| Base | Utility | Endpoint | I       |

### 6.2.3.3 Cluster ID

标识功能集ID信息详见表8。

表8 标识功能集 ID 表

| 标识符    | 名称       |
|--------|----------|
| 0x0003 | Identify |

### 6.2.3.4 数据类型

#### 6.2.3.4.1 IdentifyTypeEnum 类型

这个类型派生自enum8，详见表9。

表9 IdentifyTypeEnum 类型

| 值    | 名称               | 概述                        | 一致性 |
|------|------------------|---------------------------|-----|
| 0x00 | None             | 无任何表现                     | M   |
| 0x01 | LightOutput      | 基于灯产品的灯显示                 | M   |
| 0x02 | VisibleIndicator | 一般是一个小LED灯                | M   |
| 0x03 | AudibleBeep      | 蜂鸣装置                      | M   |
| 0x04 | Display          | 显示屏幕                      | M   |
| 0x05 | Actuator         | 将通过执行器功能传达，例如通过窗帘操作或墙内继电器 | M   |

#### 6.2.3.4.2 EffectIdentifierEnum 类型

这个类型派生自enum8，详见下表10。

表10 EffectIdentifierEnum 类型

| 值    | 名称            | 概述                                                              | 一致性 |
|------|---------------|-----------------------------------------------------------------|-----|
| 0x00 | Blink         | e. g. , 灯闪烁                                                     | M   |
| 0x01 | Breathe       | e. g. , 灯光在1秒内打开/关闭, 并重复15次                                     | M   |
| 0x02 | Okay          | e. g. , 彩色灯变绿持续1秒; 无色灯闪烁两次                                      | M   |
| 0x0B | ChannelChange | e. g. , 彩色灯变橙色持续8秒; 无色灯切换至最大亮度持续0. 5秒, 然后变为最小亮度持续7. 5秒          | M   |
| 0xFE | FinishEffect  | 在终止之前完成当前效果序列。e. g. , 如果在进行中的呼吸效果(如上所述), 首先完成当前的1秒呼吸效果, 然后终止该效果 | M   |
| 0xFF | StopEffect    | 尽快终止效果                                                          | M   |

6. 2. 3. 4. 3 EffectVariantEnum 类型

这个类型派生自enum8, 详见下表11。

表11 EffectVariantEnum 类型

| 值    | 名称      | 概述         | 一致性 |
|------|---------|------------|-----|
| 0x00 | Default | 指示默认效果已经启用 | M   |

6. 2. 3. 5 属性

6. 2. 3. 5. 1 属性列表

标识功能集属性见下表12。

表12 标识功能集属性表

| 标识符    | 名称           | 类型               | 约束条件 | 质量等级 | 默认 | 通道   | 一致性 |
|--------|--------------|------------------|------|------|----|------|-----|
| 0x0000 | IdentifyTime | uint16           | all  | -    | 0  | RWVO | M   |
| 0x0001 | IdentifyType | IdentifyTypeEnum | desc | -    | MS | RV   | M*  |

6. 2. 3. 5. 2 IdentifyTime 属性

此属性应表示端点将继续标识自身的剩余时间长度, 以秒为单位。

如果此属性设置为非0值, 则设备必须进入其识别状态, 以便向观察者指示它是多个节点和/或端点中的哪一个。建议此状态包括闪烁灯光, 周期为0. 5秒。在此状态下, IdentifyTime属性每秒递减。

如果此属性达到或设置为0值, 则设备必须终止其识别状态。

6. 2. 3. 5. 3 IdentifyType 属性

此属性应指示如何向用户展示识别状态。

此属性应包含IdentifyTypeEnum中定义的值之一。如果设备能够使用IdentifyTypeEnum中定义的其他方法之一来展示其识别状态, 则不应该使用“None”值。

6. 2. 3. 6 命令

6. 2. 3. 6. 1 命令列表

标识功能集命令见下表13。

表13 标识功能集命令表

| 标识符  | 名称             | 方向            | 返回 | 通道 | 一致性 |
|------|----------------|---------------|----|----|-----|
| 0x00 | Identify       | client→server | Y  | M  | M   |
| 0x40 | Trigger Effect | client→server | Y  | M  | O   |

### 6.2.3.6.2 Identify 命令

Identify命令用于启动或停止接收设备的自我识别功能。Identify命令必须包含以下数据字段，具体如表14所示。

表14 Identity 命令数据字段

| 标识符 | 名称           | 类型     | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|--------------|--------|------|------|----|-----|
| 0   | IdentifyTime | uint16 | all  | —    | —  | M   |

在收到Identify命令时，设备必须将Identify Time属性设置为Identify Time字段的值。这随后会根据Identify Time中的详细说明启动、继续或停止设备的识别状态。

### 6.2.3.6.3 TriggerEffect 命令

Trigger Effect命令允许支持向用户提供反馈，比如某种灯光效果。它用于在某些情况下向用户提供视觉反馈，例如在成功连接到网络时，颜色灯变为绿色。使用这个命令及其效果完全取决于实施者，在视觉反馈有用的时候可以使用，但它不同于且不替代配网过程中使用的识别机制。

Trigger Effect命令必须包含以下数据字段，具体如表15所示。

表15 TriggerEffect 命令数据字段

| 标识符 | 名称                | 类型                   | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-------------------|----------------------|------|------|----|-----|
| 0   | Effect Identifier | EffectIdentifierEnum | desc | —    | —  | M   |
| 1   | Effect Variant    | EffectVariantEnum    | desc | —    | —  | M   |

其中，

——EffectIdentifier 字段

这个字段应指示要使用的识别效果，并应包含Effect Identifier Enum中的非保留值之一。

所有Effect Identifier Enum的值都应得到支持。实施者可以偏离Effect Identifier Enum中示例的灯光效果，但应该说明在测试期间如何处理每种效果。

——EffectVariant 字段

这个字段应指示宜触发EffectIdentifier字段中指示的效果的哪个变体。如果设备不支持给定的变体，则应使用默认变体。这个字段应包含EffectVariantEnum中的一个值。

在收到此命令后，设备应执行EffectIdentifier和EffectVariant字段中指示的触发效果。如果EffectVariant字段指定了设备不支持的变体，则应执行默认变体。

## 6.2.4 组功能集

### 6.2.4.1 概述

本小节描述的是Groups功能集的功能和作用。它管理每个端点的节点范围的分组表，这是底层交互层的一部分。

在支持fabrics的网络中，由该功能集的属性或其他元素引用的组ID的范围限定为访问fabric。

Groups功能集的范围限定为端点。Groups功能集命令支持发现端点在组中的成员资格，将端点添加到组，从组中删除端点，以及从所有组中删除端点成员资格。在此< >中定义的所有命令都只应影响限定到访问fabric的组。

当支持组名时，服务端存储一个名称字符串，客户端为每个分配的组设置该名称，并在响应客户端请求时指示该名称。

请注意，对于出站命令的配置组地址是通过使用Message Layer机制实现的，其中不涉及Group Table。因此，这个功能集在这方面没有发挥作用。

### 6.2.4.2 分类

组功能集分类详见下表16。

表16 组功能集分类表

| 等级   | 角色      | 范围       | PICS 编码 |
|------|---------|----------|---------|
| Base | Utility | Endpoint | G       |

#### 6.2.4.3 Cluster ID

组功能集ID详见下表17。

表17 组功能集 ID 表

| 标识符    | 名称     |
|--------|--------|
| 0x0004 | Groups |

#### 6.2.4.4 特性

组功能集应该支持全局Feature Map属性，详见下表18。

表18 组功能集特性表

| Bit | 编码 | 特征          | 概述    |
|-----|----|-------------|-------|
| 0   | GN | Group Names | 保存分组名 |

其中，Group Names特性指的是在添加一个组时，有能力为该组存储一个名称。

#### 6.2.4.5 数据类型

Name Support Bit map类型自map8，详见下表19。

表19 Name Support Bit map 类型

| Bit | 名称         | 概述    | 一致性 |
|-----|------------|-------|-----|
| 7   | GroupNames | 保存分组名 | M   |

#### 6.2.4.6 属性

组功能集属性见下表20。

表20 组功能集属性表

| 标识符    | 名称           | 类型                | 约束条件 | 质量等级 | 默认 | 通道 | 一致性 |
|--------|--------------|-------------------|------|------|----|----|-----|
| 0x0000 | Name Support | NameSupportBitmap | desc | F    | 0  | RV | M   |

#### 6.2.4.7 Name Support 属性

这个属性提供了对组名特性是否受支持的只读访问。最重要的位，即第7位（Group Names），应等于Feature Map属性的第0位（GN特性）。所有其他位应为0。

#### 6.2.4.8 命令

##### 6.2.4.8.1 组功能集命令列表

组功能集命令见下表21。

表21 组功能集命令

| 标识符  | 名称                            | 方向            | 返回                            | 通道 | 一致性 |
|------|-------------------------------|---------------|-------------------------------|----|-----|
| 0x00 | Add Group                     | client→server | Add Group Response            | MF | M   |
| 0x01 | View Group                    | client→server | View Group Response           | OF | M   |
| 0x02 | Get Group Membership          | client→server | Get Group Membership Response | OF | M   |
| 0x03 | Remove Group                  | client→server | Remove Group Response         | MF | M   |
| 0x04 | Remove All Groups             | client→server | Y                             | MF | M   |
| 0x05 | Add Group If Identifying      | client→server | Y                             | MF | M   |
| 0x00 | Add Group Response            | client←server | N                             | —  | M   |
| 0x01 | View Group Response           | client←server | N                             | —  | M   |
| 0x02 | Get Group Membership Response | client←server | N                             | —  | M   |
| 0x03 | Remove Group Response         | client←server | N                             | —  | M   |

#### 6.2.4.8.2 Add Group 命令

##### 6.2.4.8.2.1 Add Group 命令表

Add Group命令允许客户端为服务端端点添加组成员关系，详见表22。

表22 Add Group 命令表

| 标识符 | 名称         | 类型       | 约束条件  | 质量等级 | 默认 | 一致性 |
|-----|------------|----------|-------|------|----|-----|
| 0   | Group ID   | group-id | min1  | —    | —  | M   |
| 1   | Group Name | string   | max16 | —    | —  | M   |

##### 6.2.4.8.2.2 Group ID 字段

这个字段应用于标识要添加服务端端点的组和任何关联的密钥材料。

##### 6.2.4.8.2.3 Group Name 字段

这个字段可以设置为组的可读名称。如果客户端没有为该组设置名称，则Group Name字段应设置为空字符串。

支持组名是可选的，由Feature Map和Name Support属性指示。

##### 6.2.4.8.2.4 收到后的效果

如果服务端不支持组名，则应忽略Group Name字段。在收到Add Group命令后，服务端应执行以下过程：

- 1) 如果命令字段不在约束范围内，状态应为 CONSTRAINT\_ERROR，服务端从步骤 6 继续。
- 2) 如果接收节点需要安全材料来支持组 ID，并且该组 ID 不存在该材料，状态应为 UNSUPPORTED\_ACCESS，服务端从步骤 6 继续。
- 3) 如果服务端端点是 Group ID 指示的组的成员，组名应更新（如果支持）为 Group Name，状态应为 SUCCESS，服务端从步骤 6 继续。
- 4) 如果没有可用资源来添加服务端端点的会员资格，状态应为 RESOURCE\_EXHAUSTED，服务端从步骤 6 继续。
- 5) 服务端应将服务端端点添加为 Group ID 指示的组的成员，组名应更新（如果支持）为 Group Name，状态应为 SUCCESS。
- 6) 如果由于先前的 Add Group 或 Add Group If Identifying 命令，Group ID 已经添加到组表中，并且提供了 Group Name 且服务端支持 Group Name 存储，则组表中与 Group ID 关联的

Group Name 应更新，以反映为组提供的新的 Group Name，以便后续的 View Group 命令对所有具有给定 Group ID 的组关联的端点产生相同的名称。

- 7) 如果作为单播接收到 Add Group 命令，服务端应生成一个 Add Group Response 命令，其 Status 字段设置为评估后的状态。如果作为组播接收到 Add Group 命令，服务端不应生成 Add Group Response 命令。

有关响应命令的描述，请参阅“6.3.4.9.8 Add Group Response 命令”。

#### 6.2.4.8.3 View Group 命令

View Group命令允许客户端请求服务端响应一个包含特定组名称字符串的View Group Response命令，详见下表23。

表23 View Group 命令表

| 标识符 | 名称       | 类型       | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|----------|----------|------|------|----|-----|
| 0   | Group ID | group-id | min1 | –    | –  | M   |

在收到View Group命令后，服务端应执行以下过程：

- 1) 如果命令字段不在约束范围内，状态应为 CONSTRAINT\_ERROR，并且服务端从步骤 4 继续。
- 2) 如果服务端端点是 Group ID 指示的组的成员，状态应为 SUCCESS，并且服务端从步骤 4 继续。
- 3) 否则，状态应为 NOT\_FOUND。
- 4) 如果 View Group 命令作为单播接收到，服务端应为该组生成一个 View Group Response 命令，并将 Status 字段设置为评估后的状态。如果 View Group 命令作为组播接收到，服务端不应生成 View Group Response 命令。

请参阅“6.3.4.9.9 ViewGroupResponse命令”以获取响应命令的描述。

#### 6.2.4.8.4 GetGroupMembership 命令

GetGroupMembership命令允许客户端以多种方式查询服务端端点的组成员身份，详见下表24。

表24 GetGroupMembership 命令表

| 标识符 | 名称        | 类型             | 约束条件      | 质量等级 | 默认 | 一致性 |
|-----|-----------|----------------|-----------|------|----|-----|
| 0   | GroupList | list[group-id] | all[min1] | –    | –  | M   |

当收到GetGroupMembership命令后，服务端应使用GetGroupMembershipResponse命令按照以下方式响应组成员信息：

- 如果GroupList字段为空，服务端应响应所有表示服务端端点所属组的组ID。
- 如果GroupList字段包含至少一个表示服务端端点所属组的组ID，服务端应响应每个与GroupList字段中的组匹配的表示服务端端点所属组的组ID。
- 如果GroupList字段包含一个或多个组ID，但不包含任何表示服务端端点所属组的组ID，服务端只有在命令是单播时才应响应。响应应返回一个空的GroupList字段。

#### 6.2.4.8.5 RemoveGroup 命令

RemoveGroup命令允许客户端请求服务端删除服务端端点在特定组中的成员资格（如果有），详见下表25。

表25 RemoveGroup 命令表

| 标识符 | 名称       | 类型       | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|----------|----------|------|------|----|-----|
| 0   | Group ID | group-id | min1 | –    | –  | M   |

在收到Remove Group命令后，服务端应执行以下过程：

- 1) 如果命令字段不在约束范围内，状态应为 CONSTRAINT\_ERROR，并且服务端从步骤 4 继续。

- 2) 如果服务端端点是 Group ID 指示的组的成员，服务端应移除该端点在该组中的成员资格，状态应为 SUCCESS，并且服务端从步骤 4 继续。
- 3) 否则，状态应为 NOT\_FOUND。
- 4) 如果 Remove Group 命令作为单播接收到，服务端应生成一个 Remove Group Response 命令，并将 Status 字段设置为评估后的状态。如果 Remove Group 命令作为组播接收到，服务端不应生成 Remove Group Response 命令。

请参阅“6.3.4.9.11 RemoveGroupResponse 命令”以获取响应命令的描述。

此外，如果同一端点支持 Scenes Management 功能集，则应删除该端点上与所指示的组关联的场景。

#### 6.2.4.8.6 RemoveAllGroups 命令

Remove All Groups 命令允许客户端指示服务端删除服务端端点的所有组关联。

在收到此命令后，服务端应从组表中删除服务端端点的所有组成员资格。如果 Remove All Groups 命令作为单播接收且未被抑制，则服务端应生成一个响应，将状态字段设置为 SUCCESS。

此外，如果同一端点支持场景管理功能集，除与组 ID 关联的场景外，该端点上的所有场景都应被移除。

#### 6.2.4.8.7 AddGroupIfIdentifying 命令

Add Group If Identifying 命令允许客户端在服务端端点标识自身的情况下，将该端点添加到特定组中。标识功能通过 Identify 功能集进行控制（参见章节“6.2.3 标识功能集”）。

为了正确运行 Add Group If Identifying 命令，任何支持 Groups 功能集的端点也应支持 Identify 服务端功能集。

此命令可能用于在没有配网工具的情况下帮助配置组成员关系，详见下表 26。

表 26 AddGroupIfIdentifying 命令组成员关系表

| 标识符 | 名称         | 类型       | 约束条件  | 质量等级 | 默认 | 一致性 |
|-----|------------|----------|-------|------|----|-----|
| 0   | Group ID   | group-id | min1  | -    | -  | M   |
| 1   | Group Name | string   | max16 | -    | -  | M   |

其中，

——Group ID 字段

此字段应被用来标识服务端端点将要被添加的组以及任何相关的密钥材料。

——Group Name 字段

这个字段可以设置为组的可读名称。如果客户端没有为该组设置名称，那么 Group Name 字段应该被设置为空字符串。

对组名的支持是可选的，可以通过 FeatureMap 和 NameSupport 属性来指示。

——收到后的效果

如果服务端不支持组名，那么 Group Name 字段应该被忽略。

在收到 AddGroupIfIdentifying 命令后，服务端应该执行以下过程：

- 1) 服务端验证它当前是否正在标识自身。如果服务端当前没有标识自身，状态应为 SUCCESS，并且服务端从步骤 7 继续。
- 2) 如果命令字段不在约束范围内，状态应为 CONSTRAINT\_ERROR 并且服务端从步骤 7 继续。
- 3) 如果接收节点需要安全材料来支持组 ID，并且该组 ID 不存在该材料，状态应为 UNSUPPORTED\_ACCESS 并且服务端从步骤 7 继续。
- 4) 如果服务端端点是 Group ID 指示的组的成员，状态应为 SUCCESS 并且服务端从步骤 7 继续。
- 5) 如果没有可用资源来添加服务端点的会员资格，状态应为 RESOURCE\_EXHAUSTED 并且服务端从步骤 7 继续。
- 6) 服务端应将服务端点添加为 Group ID 指示的组的成员，组名应更新(如果支持)为 Group Name，并且状态应为 SUCCESS。
- 7) 如果因为之前的 Add Group 或 Add Group If Identifying 命令已经将 Group ID 添加到 Group Table 中，并且提供了 Group Name 且服务端支持 Group Name 存储，则 Group Table 中与

Group ID 关联的 Group Name 应更新为反映为该组提供的新的 Group Name，以便后续的 View Group 命令对所有具有给定 Group ID 的组关联的端点产生相同的名称。

- 8) 如果 AddGroupIfIdentifying 命令作为单播接收，并且评估的状态不是 SUCCESS，或者如果 AddGroupIfIdentifying 命令作为单播接收并且评估的状态是 SUCCESS 但响应没有被抑制，服务端应生成一个响应，其 Status 字段设置为评估后的状态。

6.2.4.8.8 AddGroupResponse 命令

AddGroupResponse是由Groups功能集服务端发送的，以响应AddGroup命令，详见表27。

表27 AddGroupResponse 命令表

| 标识符 | 名称       | 类型       | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|----------|----------|------|------|----|-----|
| 0   | Status   | enum8    | desc | –    | –  | M   |
| 1   | Group ID | group-id | min1 | –    | –  | M   |

其中，Status字段根据AddGroup命令的“收到后的效果”部分来设置；Group ID字段设置为接收到的AddGroup命令的Group ID字段。

6.2.4.8.9 ViewGroupResponse 命令

View Group Response是由Groups功能集服务端发送，以响应View Group命令，详见下表28。

表28 ViewGroupResponse 命令表

| 标识符 | 名称         | 类型       | 约束条件  | 质量等级 | 默认 | 一致性 |
|-----|------------|----------|-------|------|----|-----|
| 0   | Status     | enum8    | desc  | –    | –  | M   |
| 1   | Group ID   | group-id | min1  | –    | –  | M   |
| 2   | Group Name | string   | max16 | –    | –  | M   |

其中，

Status字段根据View Group命令的“收到后的效果”部分来设置；

Group ID字段设置为接收到的View Group命令的Group ID字段；

Group Name字段，如果状态是成功且支持组名，那么该字段将被设置为与该组关联的组表中的组名；否则，它将被设置为空字符串。

6.2.4.8.10 GetGroupMembershipResponse 命令

GetGroupMembershipResponse命令是由Groups功能集服务端发送的，以响应GetGroupMembership命令。

GetGroupMembershipResponse命令应具有以下数据字段，详见下表29。

表29 GetGroupMembershipResponse 命令数据字段表

| 标识符 | 名称        | 类型             | 约束条件      | 质量等级 | 默认 | 一致性 |
|-----|-----------|----------------|-----------|------|----|-----|
| 0   | Capacity  | uint8          | all       | X    | –  | M   |
| 1   | GroupList | list[group-id] | all[min1] | –    | –  | M   |

其中，

——Capacity 字段

这个字段应包含节点的组表的剩余容量。以下值适用：

- a) 0-不能再添加任何组；
- b)  $0 < \text{Capacity} < 0\text{xFE} - \text{Capacity}$  表示可以添加的组的数量；
- c)  $0\text{xFE}$ -至少可以再添加一个组（确切数量未知）；
- d) null-是否还可以添加更多组未知。

——GroupList 字段

GroupList字段应包含服务端点是其成员的所有组的组ID（在接收到的GetGroupMembership命令的GroupList字段为空的情况下），或者服务端点是其成员并且其组ID包含在接收到的GetGroupMembership命令的GroupList字段中的所有组的组ID（在接收到的GetGroupMembership命令的GroupList字段不为空的情况下）。

Zigbee：如果组的总数将导致帧的最大有效载荷长度超过，则GroupList字段应仅包含适合的数量组。

6.2.4.8.11 RemoveGroupResponse 命令

RemoveGroupResponse命令是由服务端生成的，以响应收到的RemoveGroup命令，详见下表30。

表30 RemoveGroupResponse 命令表

| 标识符 | 名称       | 类型       | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|----------|----------|------|------|----|-----|
| 0   | Status   | enum8    | desc | —    | —  | M   |
| 1   | Group ID | group-id | min1 | —    | —  | M   |

其中，  
Status字段根据RemoveGroup命令的“收到后的效果”部分来设置；  
Group ID字段被设置为接收到的RemoveGroup命令的Group ID字段。

6.2.5 开/关功能集

6.2.5.1 概述

打开/关闭设备的属性和命令。

6.2.5.2 分类

开/关功能集分类详见下表31。

表31 开/关功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | 00      |

6.2.5.3 Cluster ID

开/关功能集ID详见下表32。

表32 开/关功能集 ID 表

| 标识符    | 名称     |
|--------|--------|
| 0x0006 | On/Off |

6.2.5.4 特性

6.2.5.4.1 Feature Map 位图特性

本功能集应支持如下所定义的Feature Map位图特性，详见下表33。

表33 开/关功能集 Feature Map 位图属性表

| Bit | 编码      | 特征                  | 一致性          | 概述                   |
|-----|---------|---------------------|--------------|----------------------|
| 0   | LT      | Lighting            | [!OFFONLY]   | 支持照明应用的行为            |
| 1   | DF      | Dead Front Behavior | [!OFFONLY]   | 设备有DeadFrontbehavior |
| 2   | OFFONLY | Off Only            | [! (LT DF) ] | 设备支持OffOnly功能        |

6.2.5.4.2 照明功能

这个功能集用于照明应用。  
收到Level Control功能集命令，导致On Off属性设置为FALSE，则On Time属性应设置为0。

收到Level Control功能集命令，导致On Off属性设置为TRUE，如果On Time属性值等于0，则Off Wait Time属性设置为0。

6.2.5.4.3 DeadFrontBehavior 特性

支持此特性时，当“On Off”属性为FALSE (Off)时，此服务端功能集的设备会表现出“dead front”行为。这种“dead front”行为包括：

- a) 此功能集以外的功能集会以调用和写入交互失败来响应。当处于“dead front”时，这种失败响应应带有 INVALID\_IN\_STATE 状态代码；
- b) 由于在此状态下没有定义或可用的实际值，此功能集以外的功能集会将其属性的值更改为“best-effort”值。需要支持 DF 特性的设备类型规范应该定义这些“best-effort”值是什么；
- c) 报告处理应继续生成。此类处理可以包括如上所述的“best-effort”值；
- d) 功能集以外的功能集事件生成逻辑保持不变（注意前面项目符号中可能使用的“best-effort”属性值）。

当支持此特性并且OnOff属性从TRUE变为FALSE时（例如，当收到Off命令或由于设备上的手动交互），它应开始执行此“dead front”行为。

当支持此特性并且OnOff属性从FALSE变为TRUE时（例如，当收到On命令或由于设备上的手动交互），它应停止执行此“dead front”行为。

当支持此特性，且“dead front”状态的任何变化都会导致其他功能集的属性因“dead front”而发生变化时，这些属性变化不应与属性变化相关的常规处理中跳过或省略。例如，如果由于收到Off命令，属性在“dead front”行为中值4变为null，则应处理此更改进行报告和订阅。

6.2.5.4.4 OffOnly 特性

当支持此特性，则应支持Off命令，不应支持On和Toggle命令。

此特性用于可以通过功能集接收到的Off命令关闭设备，但由于法规要求，不能用于通过功能集接收到的命令打开设备。

6.2.5.5 数据类型

6.2.5.5.1 OnOffControlBitmap 类型

该数据类型派生自map8，详见下表34。

表34 OnOffControlBitmap 类型表

| Bit | 名称               | 概述                | 一致性 |
|-----|------------------|-------------------|-----|
| 0   | AcceptOnlyWhenOn | 表示命令仅在处于打开状态时才被接受 | M   |

6.2.5.5.2 StartUpOnOffEnum 类型

该数据类型派生自map8，详见下表35。

表35 StartUpOnOffEnum 类型表

| 值 | 名称     | 概述                                                                              | 一致性 |
|---|--------|---------------------------------------------------------------------------------|-----|
| 0 | Off    | 将OnOff属性设置为FALSE                                                                | M   |
| 1 | On     | 将OnOff属性设置为TRUE                                                                 | M   |
| 2 | Toggle | 如果OnOff属性的先前值等于FALSE，请将OnOff属性设置为TRUE。如果OnOff属性的先前值等于TRUE，请将OnOff属性设置为FALSE（切换） | M   |

6.2.5.5.3 EffectIdentifierEnum 类型

该数据类型派生自map8，详见下表36。

表36 EffectIdentifierEnum 类型表

| 值    | 名称            | 概述            | 一致性 |
|------|---------------|---------------|-----|
| 0x00 | DelayedAllOff | DelayedAllOff | M   |
| 0x01 | DyingLight    | DyingLight    | M   |

## 6.2.5.5.4 DelayedAllOffEffectVariantEnum 类型

该数据类型派生自map8，详见下表37。

表37 DelayedAllOffEffectVariantEnum 类型表

| 值    | 名称                 | 概述                | 一致性 |
|------|--------------------|-------------------|-----|
| 0x00 | DelayedOffFastFade | 0.8秒后关闭           | M   |
| 0x01 | NoFade             | 无渐变               | M   |
| 0x02 | DelayedOffSlowFade | 0.8秒内变暗50%，12秒内关闭 | M   |

## 6.2.5.5.5 DyingLightEffectVariantEnum 类型

该数据类型派生自map8，详见下表38。

表38 DyingLightEffectVariantEnum 类型表

| 值    | 名称                | 概述               | 一致性 |
|------|-------------------|------------------|-----|
| 0x00 | DyingLightFadeOff | 0.5秒内变暗20%，1秒内关闭 | M   |

## 6.2.5.6 属性

## 6.2.5.6.1 关/功能集属性

关/功能集属性详见下表39。

表39 关/功能集属性表

| 标识符    | 名称                   | 类型               | 约束条件 | 质量等级 | 默认    | 通道   | 一致性 |
|--------|----------------------|------------------|------|------|-------|------|-----|
| 0x0000 | On Off               | bool             | all  | SN   | FALSE | RV   | M   |
| 0x4000 | Global Scene Control | bool             | all  | —    | TRUE  | RV   | LT  |
| 0x4001 | On Time              | uint16           | all  | —    | 0     | RWVO | LT  |
| 0x4002 | Off Wait Time        | uint16           | all  | —    | 0     | RWVO | LT  |
| 0x4003 | StartUp OnOff        | StartUpOnOffEnum | desc | XN   | MS    | RWVM | LT  |

## 6.2.5.6.2 场景表扩展

如果场景管理服务端功能集在同一端点上实现，则以下属性应成为场景表的Extension Field SetStruct的一部分。

## 6.2.5.6.3 On/Off 属性

此属性指示端点上实现的设备类型是关闭或打开，On/Off属性的值分别等于FALSE或TRUE。

## 6.2.5.6.4 GlobalSceneControl 属性

为了支持用户返回一组设备的最后设置（例如灯光的级别设置），引入了一个全局场景，该场景在设备关闭时存储，在设备打开时调用。全局场景被定义为组标识符0和场景标识符0。

定义此属性是为了防止第二个Off命令将所有设备关闭情况存储为全局场景，并防止第二条On命令通过返回全局场景破坏当前设置。

收到使On Off属性设置为TRUE的命令后，此属性应设置为TRUE，例如标准On命令、Move To Level (With On Off) 命令、Recall Scene命令或OnWithRecallGlobalScene命令。

收到Off With Effect命令后，此属性设置为FALSE。

这些概念在下图6中进行了说明。

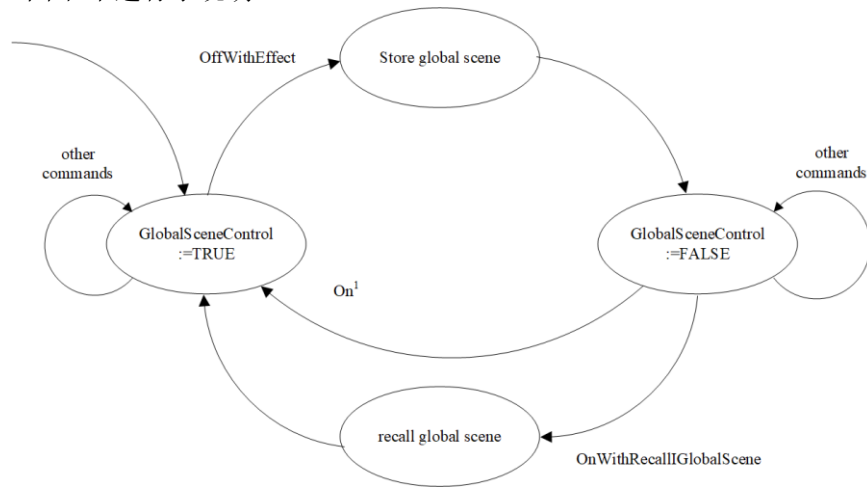


图6 使用状态图解释存储和调用全局场景功能的行为

6.2.5.6.5 OnTime 属性

此属性指定使用On With Timed Off命令时，在自动转换为Off状态之前，应保持On状态的时间长度（以1/10秒为单位）。此属性可以随时写入，但写入值仅在处于定时打开状态时有效。

6.2.5.6.6 OffWaitTime 属性

此属性指定关闭状态的时间长度（以1/10秒为单位），以防止另一个On With Timed Off命令将服务端恢复到打开状态（例如，离开房间时，灯关闭，但占用传感器检测到人未离开并试图重新打开灯）。此属性可以在任何时候写入，但写入值仅在处于定时打开状态，然后转换为延迟关闭状态或处于延迟关闭状态时有效。

6.2.5.6.7 StartUpOnOff 属性

此属性应定义设备在通电时的启动行为，此状态应反映在On Off属性中。如果该值为null，则On Off属性将设置为其先前的值。否则，该行为将在StartUpOnOffEnum的表中定义。

此行为不适用于与OTA相关的重启。OTA重启后，On Off属性应恢复到重启前的值。

6.2.5.7 命令

6.2.5.7.1 开/关功能集命令

该功能集命令详见下表40。

表40 开/关功能集命令表

| 标识符  | 名称                      | 方向            | 返回 | 通道 | 一致性      |
|------|-------------------------|---------------|----|----|----------|
| 0x00 | Off                     | client→server | Y  | 0  | M        |
| 0x01 | On                      | client→server | Y  | 0  | !OFFONLY |
| 0x02 | Toggle                  | client→server | Y  | 0  | !OFFONLY |
| 0x40 | OffWithEffect           | client→server | Y  | 0  | LT       |
| 0x41 | OnWithRecallGlobalScene | client→server | Y  | 0  | LT       |
| 0x42 | On With Timed Off       | client→server | Y  | 0  | LT       |

6.2.5.7.2 Off 命令

收到Off命令后，服务端应将OnOff属性设置为FALSE。此外，当支持OnTime属性时，服务端应将OnTime属性设置为0。

6.2.5.7.3 On 命令

如果支持OffOnly功能，在收到on命令后，应发送UNSUPPORTED\_COMMAND故障状态响应。否则，在收到on命令后，服务端应将OnOff属性设置为TRUE。

此外，当OnTime和OffWaitTime属性都支持时，如果OnTime属性的值等于0，服务端应将OffWaitTime属性设置为0。

6.2.5.7.4 Toggle 命令

如果支持OffOnly功能，在收到Toggle命令后，应发送UNSUPPORTED\_COMMAND故障状态响应。否则，在收到Toggle命令后，如果OnOff属性的值等于FALSE，则服务端应将OnOff属性设置为TRUE，否则，服务端应将OnOff属性设置成FALSE。

此外，当同时支持OnTime和OffWaitTime属性时，如果OnOff属性的值等于FALSE，并且OnTime属性的值为0，则服务端应将OffWaitTime特性设置为0。如果OnOff属性的值等于TRUE，则服务端应将OnTime属性设置为0。

6.2.5.7.5 OffWithEffect 命令

OffWithEffect命令允许使用增强的渐变方式关闭设备。

OffWithEffect命令应具有数据字段，详见表41。

表41 OffWithEffect 命令的数据字段表

| 标识符 | 名称               | 类型                   | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|------------------|----------------------|------|------|----|-----|
| 0   | EffectIdentifier | EffectIdentifierEnum | desc | —    | —  | M   |
| 1   | EffectVariant    | enum8                | desc | —    | 0  | M   |

其中，

- EffectIdentifier 字段指定关闭设备时使用的渐变效果。此字段应包含EffectIdentifierEnum中列出的非保留值之一；
- EffectVariant字段用于指示应触发EffectIdentifier字段中指示的效果的变体。如果服务端不支持给定的变体，则应使用默认变体。此字段取决于EffectIdentifier字段的值，并且应包含DelayedAllOffEffectVariantEnum或DyingLightEffectVariantEnum枚举中列出的非保留值之一。

收到OffWithEffect命令后，服务端应检查GlobalSceneControl属性的值。

如果GlobalSceneControl属性等于TRUE，则服务端应将其设置存储在其全局场景中，然后将GlobalSceneConcontrol属性设置为FALSE，将OnOff属性设置为FALSE，如果支持OnTime属性，则将OnTime属性设置为0。

如果GlobalSceneControl属性等于FALSE，则服务端应仅将OnOff属性设置为FALSE。

6.2.5.7.6 OnWithRecallGlobalScene 命令

此命令允许在设备关闭时调用设置。

收到OnWithRecallGlobalScene命令后，如果GlobalSceneControl属性等于TRUE，服务端应丢弃该命令。

如果GlobalSceneControl属性等于FALSE，则同一段点上的场景管理功能集服务端应调用其全局场景，相应的更新OnOff属性。然后，OnOff服务端应将GlobalSceneControl属性设置为TRUE。

此外，当OnTime和OffWaitTime属性都支持时，如果OnTime属性的值等于0，服务端应将OffWaitTime属性设置为0。

6.2.5.7.7 On With Timed Off 命令

此命令允许设备在特定持续时间内打开，并具有保护关闭持续时间，以便如果随后关闭设备，则在此期间收到的进一步On With Timed Off命令将被阻止重新打开设备。服务端打开时收到的进一步On With Timed Off命令将更新设备打开的时间。

On With Timed Off命令应具有数据字段，详见下表42。

表42 OnWithTimedOff 命令数据字段表

| 标识符 | 名称           | 类型                 | 约束条件      | 质量等级 | 默认 | 一致性 |
|-----|--------------|--------------------|-----------|------|----|-----|
| 0   | OnOffControl | OnOffControlBitmap | 0to1      | —    | —  | M   |
| 1   | OnTime       | uint16             | max0xFFFE | —    | —  | M   |
| 2   | OffWaitTime  | uint16             | max0xFFFE | —    | —  | M   |

其中，

- OnOffControl字段包含有关如何操作服务端的信息；
- AcceptOnlyWhenOn位，指定是无条件处理On With Timed Off命令，或仅在OnOff属性等于TRUE时处理。如果此子字段设置为1，则只有当OnOff属性等于TRUE时，才会接受On With Timed Off命令。如果此子字段设置为0，则应无条件处理On With Timed Off命令；
- OnTime字段用于调整OnTime属性的值；
- OffWaitTime字段用于调整OffWaitTime属性的值。

收到此命令后，如果OnOffControl字段的AcceptOnlyWhenOn子字段设置为1，并且OnOff属性的值等于FALSE，则应丢弃该命令。

如果OffWaitTime属性的值大于零，OnOff属性的值等于FALSE，则服务端应将OffWaitTime设置为OffWaitTime属性和OffWaitTime字段中指定值的最小值。

在所有其他情况下，服务端应将OnTime属性设置为OnTime属性和OnTime字段中指定值的最大值，将OffWaitTime属性设置为由OffWaitTime字段指定的值，并将OnOff属性设置为TRUE。

如果OnTime和OffWaitTime属性的值都不等于0xFFFF，则服务端应每 $\lceil 1/10 \rceil^{\text{th}}$ 秒更新一次这些属性，直到OnTime和OffWaitTime属性都等于0，如下所示：

如果OnOff属性的值等于TRUE且OnTime属性的值大于零，则服务端应递减OnTime属性。如果OnTime属性的值达到0，服务端应将OffWaitTime和OnOff属性分别设置为0和FALSE。

如果OnOff属性的值等于FALSE，并且OffWaitTime属性的值大于零，则服务端应递减OffWaitTime属性的值。如果OffWaitTime属性的值达到0，服务端应终止更新。

6.2.5.8 状态描述

图7“On/Off功能集操作状态机”中说明了开/关功能集相对于On、Off和On With Timed Off命令的操作。在图7中，值X和Y分别对应于On With Timed Off命令的OnTime和OffWaitTime字段。在TimedOn状态下，OnTime属性每 $\lceil 1/10 \rceil^{\text{th}}$ 秒递减一次，除非其值等于0xFFFF。同样，在延迟关闭状态下，OffWaitTime属性每 $\lceil 1/10 \rceil^{\text{th}}$ 秒递减一次，除非其值等于0xFFFF。

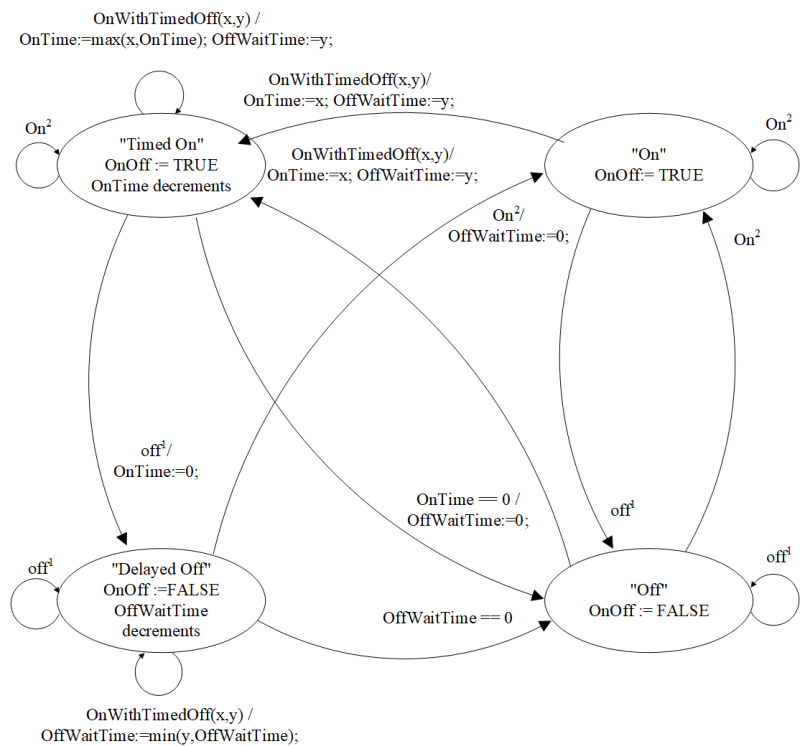


图7 On/Off 功能集操作状态机

6.2.6 级别控制功能集

6.2.6.1 概述

此功能集提供了一个接口，用户可以通过该接口控制设备的可调节属性。例如灯光的亮度、门的关闭程度或加热器的功率输出。

6.2.6.2 分类

级别控制功能集分类，详见下表43。

表43 级别控制功能集分类表

| 等级   | 角色          | 范围       | PICS编码 |
|------|-------------|----------|--------|
| Base | Application | Endpoint | LVL    |

6.2.6.3 Cluster ID

这个基础功能集（base cluster）规范可以用于通用的级别控制；但仍建议派生另一个功能集（cluster），以更好地定义应用和领域的需求。如果一个或多个派生功能集（cluster）标识符和基础标识符存在于设备端点，则它们都应表示设备级控制的单个实例，详见下表44。

表44 级别控制功能集 ID 表

| 标识符    | 名称           |
|--------|--------------|
| 0x0008 | LevelControl |

6.2.6.4 特性

6.2.6.4.1 Feature Map 位图属性

该功能集应支持如下所定义的Feature Map位图属性，详见下表45。

表45 别控制功能集 Feature Map 位图属性表

| Bit | 编码 | 特征        | 默认 | 一致性 | 概述                |
|-----|----|-----------|----|-----|-------------------|
| 0   | 00 | On Off    | 1  | 0   | 依赖于On/Off cluster |
| 1   | LT | Lighting  | 0  | 0   | 支持照明应用的行为         |
| 2   | FQ | Frequency | 0  | P   | 支持频率属性和行为         |

以下章节详细描述了每个特性，更多细节可以在规范中找到。

6.2.6.4.2 On/Off 特性

6.2.6.4.2.1 概述

对于许多应用，此cluster与On/Off cluster之间需要有紧密的关系。本节描述了同时实现该服务端cluster和On/Off服务端cluster的端点所需的依赖关系。在Feature Map中引入On/Off特性位之前，此cluster与On/Off cluster之间就已存在依赖关系。

On/Off cluster的OnOff属性和LevelControlcluster的CurrentLevel属性本质上是独立的变量，因为它们位于不同的cluster中。然而，当这两个cluster都在同一端点上部署时，可以在它们之间引入依赖关系。如果需要，可以提供相关设施来引入这些依赖关系。

6.2.6.4.2.2 On/Off 命令对 CurrentLevel 属性的影响

OnLevel属性决定了On/Offcluster的命令是否对CurrentLevel属性产生永久性影响。如果该属性已定义（即已实现且不为null），则它们会产生永久性影响；否则，不会产生永久性影响。由于渐变效果的存在，一定会有暂时的影响。

当接收到On/Offcluster的各种命令时，对LevelControlcluster的影响如表“On/Off命令”所述。在该表以及在整个cluster规范中，'level'表示CurrentLevel属性的值，详见下表46。

表46 On/Off 命令与级别控制关联的影响

| 命令     | 收到命令后的动作                                                                                                                 |
|--------|--------------------------------------------------------------------------------------------------------------------------|
| On     | 临时存储CurrentLevel。<br>将CurrentLevel设置为设备允许的最小级别。在OnOffTransitionTime时间段内，将CurrentLevel更改为OnLevel，如果OnLevel未定义，则更改为已存储的级别。 |
| Off    | 临时存储CurrentLevel。<br>在OnOffTransitionTime时间段内，将CurrentLevel更改为设备允许的最小级别。如果OnLevel未定义，则将CurrentLevel设置为已存储的级别。            |
| Toggle | 如果OnOff属性的值为FALSE，则按照On命令的操作进行处理。否则，按照Off命令的操作进行处理。                                                                      |

上表中描述的操作的意图是，在On、Off或Toggle命令发出之前生效的CurrentLevel应该在On、Off或Toggle命令转换（transition）完成后恢复。如果在转换（transition）完成之前接收到另一个此类命令，原本存储的CurrentLevel应该保留并在转换（transition）完成后恢复。

6.2.6.4.2.3 级别控制命令对 OnOff 属性的影响

在LevelControlcluster中提供了两组命令。这两组命令完全相同，唯一的区别在于第一组（MoveToLevel、Move和Step命令）不得作用于OnOff属性，而第二组（带On/Off变体的命令）必须能作用于OnOff属性。

第一组命令用于保持CurrentLevel和OnOff属性的独立性，因此改变CurrentLevel不会影响OnOff属性。举例来说，这类似于一个带有独立静音按钮的音量控制器，或者一个“可通过旋钮调节亮度级别，并可通过按键开关灯”的灯光调光器。

第二组命令用于关联CurrentLevel和OnOff属性。当级别降低到最低值时，OnOff属性会自动设置为FALSE，当级别增加到最低值以上时，OnOff属性会自动设置为TRUE。举例来说，这类似于一个没有独立开关的灯光调光器。

对于Stop命令，StopWithOnOff仅用于对称性设计，以便轻松选择其中一组命令——这两种Stop命令是相同的，因为对On/Off的依赖性取决于正在停止的原始命令。

6.2.6.4.2.4 级别控制命令的效果取决于 On/Off 属性

在Options属性引入之前，除了后缀为“withOn/Off”的命令外，所有命令在同一端点的On/Offcluster的OnOff属性为FALSE时都不会产生任何效果。即使On/Off（00）功能集位被设置为零，这一点仍然成立。为了使这些命令能够正常工作，请参阅“6.3.6.7.10 Options属性”。

6.2.6.4.2.5 GlobalSceneControl 和带有 On/Off 的命令

如果接收到MoveToLevel（WithOnOff）、Move（WithOnOff）或Step（WithOnOff）命令，这些命令导致On/Offcluster的OnOff属性的值发生变化，则On/Offcluster的GlobalSceneControl属性应根据GlobalSceneControl章节的规定进行更新。

6.2.6.4.3 照明特性

- 该特性支持光源的级别控制接口。
- 对于CurrentLevel属性：
- 0x00 值不得使用。
  - 0x01 值应表示设备可以达到的最小级别。
  - 0xFE 值应表示设备可以达到的最大级别。null 值应表示未定义的值。
  - 所有其他值是应用特定的级别，表示从最小级别到最大级别的渐变。

6.2.6.4.4 频率特性

频率特性是暂定的。

6.2.6.5 数据类型

6.2.6.5.1 OptionsBitmap 类型

该数据类型派生自map8，详见下表47。

表47 OptionsBitmap 类型表

| Bit | 名称                     | 概述                     | 一致性   |
|-----|------------------------|------------------------|-------|
| 0   | ExecuteIfOff           | 依赖于On/Offcluster       | LT 00 |
| 1   | CoupleColorTempToLevel | 依赖于ColorControlcluster | L     |

- 其中，
- ExecuteIfOff位，该位指示此功能集与On/Off功能集是否存在依赖关系；
  - CoupleColorTempToLevel位，该位指示此功能集与ColorControl功能集是否存在依赖关系。

6.2.6.5.2 MoveModeEnum 类型

该数据类型派生自enum8，详见下表48。

表48 MoveModeEnum 类型表

| 值 | 名称   | 概述   | 一致性 |
|---|------|------|-----|
| 0 | Up   | 级别调高 | M   |
| 1 | Down | 级别调低 | M   |

6.2.6.5.3 StepModeEnum 类型

该数据类型派生自enum8，详见下表49。

表49 StepModeEnum 类型表

| 值 | 名称   | 概述   | 一致性 |
|---|------|------|-----|
| 0 | Up   | 向上调节 | M   |
| 1 | Down | 向下调节 | M   |

### 6.2.6.6 属性

#### 6.2.6.6.1 级别控制功能集属性

级别控制功能集属性详见下表50。

表50 级别控制功能集属性表

| 标识符    | 名称                  | 类型            | 约束条件                       | 质量等级 | 默认   | 通道   | 一致性   |
|--------|---------------------|---------------|----------------------------|------|------|------|-------|
| 0x0000 | CurrentLevel        | uint8         | MinLeveltoMaxLevel         | SNXQ | null | RV   | M     |
| 0x0001 | RemainingTime       | uint16        | all                        | Q    | 0    | RV   | LT    |
| 0x0002 | MinLevel            | uint8         | 1to254                     | —    | 1    | RV   | [LT]  |
| 0x0002 | MinLevel            | uint8         | max254                     | —    | 0    | RV   | [!LT] |
| 0x0003 | MaxLevel            | uint8         | MinLevelto254              | —    | 254  | RV   | 0     |
| 0x0004 | CurrentFrequency    | uint16        | MinFrequencytoMaxFrequency | PSQ  | 0    | RV   | FQ    |
| 0x0005 | MinFrequency        | uint16        | all                        | —    | 0    | RV   | FQ    |
| 0x0006 | MaxFrequency        | uint16        | minMinFrequency            | —    | 0    | RV   | FQ    |
| 0x0010 | OnOffTransitionTime | uint16        | all                        | —    | 0    | RWVO | 0     |
| 0x0011 | OnLevel             | uint8         | MinLeveltoMaxLevel         | X    | null | RWVO | M     |
| 0x0012 | OnTransitionTime    | uint16        | all                        | X    | null | RWVO | 0     |
| 0x0013 | OffTransitionTime   | uint16        | all                        | X    | null | RWVO | 0     |
| 0x0014 | DefaultMoveRate     | uint8         | min1                       | X    | MS   | RWVO | 0     |
| 0x000F | Options             | OptionsBitmap | desc                       | —    | 0    | RWVO | M     |
| 0x4000 | StartUpCurrentLevel | uint8         | desc                       | XN   | MS   | RWVM | LT    |

#### 6.2.6.6.2 SceneTable 扩展

如果场景管理服务端功能集(cluster)在同一端点实现,则以下属性应作为场景表(SceneTable)的ExtensionFieldSetStruct的一部分。如果使用隐式形式的ExtensionFieldSetStruct,则AttributeValueList中属性的顺序应按给定顺序排列,即列为第1位的属性首先添加:

- CurrentLevel
- CurrentFrequency

在场景表(SceneTable)中,设备不支持的属性(根据FeatureMap属性)虽不被采用,仍应在场景中列出。对于超出相应MoveTo<Target>命令接受值范围的扩展字段集值,例如与MoveToLevel命令对应的CurrentLevel属性的null值,应被忽略。

#### 6.2.6.6.3 CurrentLevel 属性

该属性应表示设备的当前级别。级别的含义依赖于设备。只有在以下情况下,对该属性的更改才应标记为可报告:

- 每秒最多一次,或
- 在调节/转换结束时,或
- 当其值从null变为任何其他值,或反之亦然。

#### 6.2.6.6.4 RemainingTime 属性

该属性应表示当前命令完成前的剩余时间,单位为1/10秒。只有在以下情况下,对该属性的更改才应标记为可报告:

- 当其值从0变为大于10的任何值时,或
- 当其值发生变化,且变化量大于10,由命令调用引起时,或
- 当其值变为0时。

对于具有转换时间或转换时间小于1秒的命令,不应报告该属性的更改。

由于在常规倒计时期间此属性不会被报告,因此客户端不应依赖该属性的报告来跟踪剩余时长。

#### 6.2.6.6.5 MinLevel 属性

该属性应表示可以分配的CurrentLevel的最小值。

#### 6.2.6.6.6 MaxLevel 属性

该属性应表示可以分配的CurrentLevel的最大值。

#### 6.2.6.6.7 CurrentFrequency 属性

该属性应表示设备在CurrentLevel下的频率。CurrentFrequency为0表示未知。只有在以下情况下，对该属性的更改才应标记为可报告：

- a) 每秒最多一次，或
- b) 在调节/转换开始时，或
- c) 在调节/转换结束时。

#### 6.2.6.6.8 MinFrequency 属性

该属性应表示可以分配的CurrentFrequency的最小值。MinFrequency应小于或等于MaxFrequency。值为0表示未定义。

#### 6.2.6.6.9 MaxFrequency 属性

该属性应表示可以分配的CurrentFrequency的最大值。MaxFrequency应大于或等于MinFrequency。值为0表示未定义。

#### 6.2.6.6.10 Options 属性

该属性应表示设备的选定选项。

Options属性是一个位图(bitmap)数据类型，用于确定某些功能集(cluster)命令的默认行为。

每个依赖于Options属性的命令应首先构建一个临时的Options位图(bitmap)，该位图在命令处理期间生效。临时Options位图(bitmap)的格式和意义与Options属性相同，但包含任何可能被命令字段覆盖的位。

该属性仅在设备调试过程中进行更改。

如果满足以下所有条件，命令执行应在Options处理后停止：

- a) 命令是“无 On/Off”命令之一：Move、MoveToLevel、Step 或 Stop；
- b) On/Offcluster 存在于与该功能集(cluster)相同的端点上；
- c) 该端点的 On/Offcluster 的 OnOff 属性为 FALSE；
- d) ExecuteIfOff 位的值为 0。

其中，

- ExecuteIfOff位，如果该位被设置，当On/Offcluster的OnOff属性为FALSE时，本功能集(cluster)中的命令仍会执行，并可能更改CurrentLevel属性。
- CoupleColorTempToLevel位，如果该位被设置，CurrentLevel属性的更改应与ColorControlcluster中设置的色温相关联，当不支持Lighting特性时，该位应为0并被忽略。

#### 6.2.6.6.11 OnOffTransitionTime 属性

该属性应表示在接收到On/Offcluster上的On或Off命令时，转换到目标级别所需的时间。该时间以1/10秒为单位进行指定。

实际所需时间应尽可能接近OnOffTransitionTime，但需注意，如果设备无法以可变速率调节，则OnOffTransitionTime属性不应被实现。

#### 6.2.6.6.12 OnLevel 属性

作为处理On/Offcluster命令的结果，当同一端点的On/Offcluster中的OnOff属性被设置为TRUE时，OnLevel属性应指示CurrentLevel被设置的值。如果OnLevel属性没有实现，或设置为null值，则不产生任何效果。有关详细信息，请参见章节“6.2.6.4.2.2 On/Off命令对CurrentLevel属性的影响”。

OnLevel是一个强制性字段，之前该字段可能不存在或为可选字段。实施者(Implementers)应注意：较旧的设备可能没有实现该字段。

#### 6.2.6.6.13 OnTransitionTime 属性

该属性应表示在接收到On/Offcluster上的On命令时,将当前级别从最小级别转化到最大级别所需的时间。该时间以1/10秒为单位进行指定。如果此属性未实现或包含null值,则应使用OnOffTransitionTime作为替代。

6.2.6.6.14 OffTransitionTime 属性

该属性应表示在接收到On/Offcluster上的Off命令时,将当前级别从最大级别转换到最小级别所需的时间。该时间以1/10秒为单位进行指定。如果此属性未实现或包含null值,则应使用OnOffTransitionTime作为替代。

6.2.6.6.15 DefaultMoveRate 属性

该属性应表示在接收到Move命令且Rate参数为null值时的调节速率,以每秒单位表示。

6.2.6.6.16 StartUpCurrentLevel 属性

该属性应表示设备通电时的预设启动级别,并且该级别应反映在CurrentLevel属性中。StartUpCurrentLevel属性的值详见下表51。

表51 StartUpCurrentLevel 属性表

| 值           | 开机后的动作                     |
|-------------|----------------------------|
| 0           | 将CurrentLevel属性设置为设备允许的最小值 |
| null        | 将CurrentLevel属性设置为其先前的值    |
| othervalues | 将CurrentLevel属性设置为该值       |

此行为不适用于与OTA相关的重启。OTA重启后,CurrentLevel属性应恢复到重启前的值。

6.2.6.7 命令

6.2.6.7.1 级别控制功能集命令

该功能集命令详见下表52。

表52 级别控制功能集命令表

| 标识符  | 名称                     | 方向            | 返回 | 通道 | 一致性 |
|------|------------------------|---------------|----|----|-----|
| 0x00 | MoveToLevel            | client→server | Y  | 0  | M   |
| 0x01 | Move                   | client→server | Y  | 0  | M   |
| 0x02 | Step                   | client→server | Y  | 0  | M   |
| 0x03 | Stop                   | client→server | Y  | 0  | M   |
| 0x04 | MoveToLevelWithOnOff   | client→server | Y  | 0  | M   |
| 0x05 | MoveWithOnOff          | client→server | Y  | 0  | M   |
| 0x06 | StepWithOnOff          | client→server | Y  | 0  | M   |
| 0x07 | StopWithOnOff          | client→server | Y  | 0  | M   |
| 0x08 | MoveToClosestFrequency | client→server | Y  | 0  | FQ  |

6.2.6.7.2 MoveToLevel 命令

MoveToLevel命令详见下表53。

表53 MoveToLevel 命令表

| 标识符 | 名称              | 类型            | 约束条件   | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|--------|------|----|-----|
| 0   | Level           | uint8         | max254 | —    | —  | M   |
| 1   | TransitionTime  | uint16        | all    | X    | —  | M   |
| 2   | OptionsMask     | OptionsBitmap | desc   | —    | 0  | M   |
| 3   | OptionsOverride | OptionsBitmap | desc   | —    | 0  | M   |

其中，OptionsMask和OptionsOverride字段必须同时存在。默认值将用于解释来自旧设备的缺失字段。应根据Options属性、OptionsMask和OptionsOverride字段创建临时的Options位图(bitmap)。每个位图的位应按以下规则确定：

Options属性中的每个位决定临时Options位图中的对应位，除非OptionsMask字段存在且相应位设置为1，在这种情况下，OptionsOverride字段中的对应位应决定临时Options位图中的对应位。

attribute。生成的临时Options位图应按照Options属性中定义的方式进行处理。

在接收到此命令时，设备应从当前级别调节到Level字段中给定的值。‘级别(level)’的含义取决于设备类型——例如，对于灯光来说，可能表示亮度级别。

该级别变化应尽可能连续，即非阶跃函数，且调节到新级别所需的时间应等于TransitionTime字段的值（单位为十分之一秒），或者尽可能接近该值。

如果TransitionTime字段的值为null，则调节到新级别所需的时间应由OnOffTransitionTime属性来确定。如果OnOffTransitionTime（可选属性）不存在，设备应尽快调节到新级别。

如果设备无法以可变速率进行调节，则可以忽略TransitionTime字段。

6.2.6.7.3 Move 命令

Move命令详见下表54。

表54 Move 命令表

| 标识符 | 名称              | 类型            | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|------|------|----|-----|
| 0   | MoveMode        | MoveModeEnum  | desc | —    | —  | M   |
| 1   | Rate            | uint8         | all  | X    | —  | M   |
| 2   | OptionsMask     | OptionsBitmap | desc | —    | 0  | M   |
| 3   | OptionsOverride | OptionsBitmap | desc | —    | 0  | M   |

其中，

- MoveMode字段应为MoveModeEnum中的非保留值之一。
- Rate字段应指示调节速率，单位为每秒的单位。实际的调节速率应尽可能接近该速率，具体取决于设备的能力。如果Rate字段为null，则应使用DefaultMoveRate属性的值（如果该属性得到支持且其值不为null）。如果Rate字段为null且DefaultMoveRate属性不被支持或其值为null，则设备应尽可能以最快速度调节。如果设备无法以可变速率调节，则可以忽略此字段。

在接收到此命令时，如果Rate字段的值为零，则该命令无效，设备应返回状态码INVALID\_COMMAND。

否则，设备应首先创建并处理一个临时的Options位图，如同MoveToLevel命令中的描述。设备应从当前级别开始，在Up或Down方向上以连续的方式调节，详见下表55。

表55 MoveMode 字段

| MoveMode | 收到命令后的动作                               |
|----------|----------------------------------------|
| Up       | 以Rate字段给定的速率增加设备的级别。如果级别达到设备允许的最大值，则停止 |
| Down     | 以Rate字段给定的速率减少设备的级别。如果级别达到设备允许的最小值，则停止 |

6.2.6.7.4 Step 命令

Step命令详见下表56。

表56 Step 命令表

| 标识符 | 名称              | 类型            | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|------|------|----|-----|
| 0   | StepMode        | StepModeEnum  | desc | —    | —  | M   |
| 1   | StepSize        | uint8         | all  | —    | —  | M   |
| 2   | TransitionTime  | uint16        | all  | X    | —  | M   |
| 3   | OptionsMask     | OptionsBitmap | desc | —    | 0  | M   |
| 4   | OptionsOverride | OptionsBitmap | desc | —    | 0  | M   |

其中，

- StepMode字段应为StepModeEnum中的非保留值。
- StepSize字段应表示对CurrentLevel的变化量。
- TransitionTime字段应表示执行步进所需的时间，以1/10秒为单位。步进是CurrentLevel的变化，单位为StepSize。实际所需时间应尽可能接近此值。如果TransitionTime字段的值为null，则设备应尽最大可能尽快调节。如果设备无法以可变速率调节，则TransitionTime字段可以被忽略。

在接收到此命令时，如果StepSize字段的值为零，则该命令无效，并应返回一个状态码为INVALID\_COMMAND的响应。

否则，设备应首先创建并处理一个临时的Options位图，如MoveToLevel命令中所述。在接收到此命令后，设备应根据说明从其当前级别开始向上或向下调节，详见下表57。

表57 设备接收到命令后的说明表

| StepMode | 收到命令后的动作                                                             |
|----------|----------------------------------------------------------------------|
| Up       | 将CurrentLevel增加StepSize单位，或者如果在此过程中达到设备允许的最大级别，则停止。在后者情况下，转换时间应按比例减少 |
| Down     | 将CurrentLevel减少StepSize单位，或者如果在此过程中达到设备允许的最小级别，则停止。在后者情况下，转换时间应按比例减少 |

6.2.6.7.5 Stop 命令

Stop命令信息详见下表58。

表58 Stop 命令表

| 标识符 | 名称              | 类型            | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|------|------|----|-----|
| 0   | OptionsMask     | OptionsBitmap | desc | —    | 0  | M   |
| 1   | OptionsOverride | OptionsBitmap | desc | —    | 0  | M   |

在接收到此命令时，设备应首先创建并处理一个临时的Options位图，如MoveToLevel命令中所述。

接收到此命令后，任何正在执行的MoveToLevel、Move或Step命令（及其带On/Off的变体）都应终止。CurrentLevel的值应保持在接收到Stop命令时的值，RemainingTime应设置为零。

此命令在章节“命令”列表中有两个条目：一个是针对MoveToLevel、Move和Step命令，另一个是针对它们的“withOn/Off”变体。这样做完全是为了对称性，允许轻松选择一个命令集。Stop命令是相同的，因为是否依赖于On/Off由原始命令的停止方式决定。

6.2.6.7.6 MoveToClosestFrequency 命令

MoveToClosestFrequency命令详见下表59。

表59 MoveToClosestFrequency 命令表

| 标识符 | 名称        | 类型     | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------|--------|------|------|----|-----|
| 0   | Frequency | uint16 | all  | —    | 0  | M   |

接收到此命令后，设备应将其当前频率更改为请求的频率，或调整到最接近的可以生成的频率。如果设备无法逼近该频率，则应返回默认响应，错误码为CONSTRAINT\_ERROR。是否可以将请求的频率近似为受支持的频率是由制造商决定的。

6.2.6.7.7 'WithOn/Off' 命令

MoveToLevelWithOnOff、MoveWithOnOff和StepWithOnOff命令与MoveToLevel、Move和Step命令的字段相同。它们的效果也相同，唯一的不同在于以下几点。

在执行任何将CurrentLevel属性设置为大于设备允许的最小级别的命令之前，如果实现了On/Offcluster，则同一端点的On/Off属性应设置为TRUE（即“开”）。

如果任何命令将CurrentLevel属性设置为设备允许的最小级别，则同一端点的On/Off属性应设置为FALSE（即“关”）。

StopWithOnOff命令与Stop命令的字段完全相同。两个Stop命令是相同的，因为On/Off的依赖关系由原始命令的停止方式决定。

6.2.6.8 灯光状态变更表

在此表格中列出了当LevelControl和On/Offcluster位于同一端点，并且启用了Lighting功能时的状态变化示例。该表展示了在不同命令和设备状态之间的转换。

- EiO-Options 属性中的 ExecuteIfOff 字段
  - OnOff-On/Offcluster 的属性值：FALSE=‘Off’，TRUE=‘On’MIN-MinLevel（最小亮度级别）
  - MAX-MaxLevel（最大亮度级别）
  - MID-MinLevel 和 MaxLevel 之间的中点（亮度的中值）
- 照明设备状态变化详见下表60。

表60 照明设备状态变化图

| 当前状态 | EiO | On/Off | 物理设备              | 命令前后                               | 当前状态 | On/Off | 物理设备                    | 设备输出结果             |
|------|-----|--------|-------------------|------------------------------------|------|--------|-------------------------|--------------------|
| any  | 0   | FALSE  | Off               | MoveToLevel（l=MID，t=sec）           | same | FALSE  | Off                     | 依然保持关闭             |
| any  | 0   | FALSE  | Off               | MoveToLevelWithOnOff（l=MID，t=2sec  | MID  | TRUE   | On（mid-pointbrightness） | 开启并调整输出水平，或保持在一半   |
| any  | 1   | FALSE  | Off               | MoveToLevel（l=MID，t=2sec）          | MID  | FALSE  | Off                     | 依然保持关闭             |
| any  | 1   | FALSE  | Off               | MoveToLevelWithOnOff（l=MID，t=2sec） | MID  | TRUE   | On                      | 开启并调整输出水平，或保持在一半   |
| any  | 1   | FALSE  | Off               | Move（up，rate=64/s）                 | MAX  | FALSE  | Off                     | 依然保持关闭             |
| any  | 1   | FALSE  | Off               | MoveWithOnOff（up，rate=64/s）        | MAX  | TRUE   | On                      | 开启并调整输出水平，或保持在最大亮度 |
| any  | 1   | FALSE  | Off               | MoveWithOnOff（down，rate=64/s）      | MIN  | FALSE  | Off                     | 依然保持关闭             |
| any  | any | TRUE   | On（anybrightness） | MoveToLevelWithOnOff（l=MID，t=2sec） | MID  | TRUE   | On（mid-pointbrightness） | 输出水平调整到或保持在一半      |
| any  | any | TRUE   | On（anybrightness） | MoveWithOnOff（up，rate=64/s）        | MAX  | TRUE   | On（fullbrightness）      | 输出水平调整到或保持在最大亮度    |
| any  | any | TRUE   | On（anybrightness） | Move（down，rate=64/s）               | MIN  | TRUE   | On（atminimumbrightness） | 输出水平调整到最小亮度        |
| any  | any | TRUE   | On（anybrightness） | MoveWithOnOff（down，rate=64/s）      | MIN  | FALSE  | Off                     | 输出水平调整到关闭          |

6.2.7 布尔状态功能集

6.2.7.1 概述

该功能集提供了一个布尔状态的接口。

6.2.7.2 分类

布尔状态功能集分类详见下图66。

表61 布尔状态功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | BOOL    |

#### 6.2.7.3 Cluster ID

布尔状态功能集ID详见下表62。

表62 布尔状态功能集 ID 表

| 标识符    | 名称           |
|--------|--------------|
| 0x0045 | BooleanState |

#### 6.2.7.4 属性

布尔状态功能集属性详见下表63。

表63 布尔状态功能集属性表

| 标识符    | 名称         | 类型   | 约束条件 | 质量等级 | 默认 | 通道 | 一致性 |
|--------|------------|------|------|------|----|----|-----|
| 0x0000 | StateValue | bool | —    | P    | —  | RV | M   |

其中，StateValue属性表示布尔态。此布尔态的语义由使用此功能集的设备类型定义。例如，在一种“接触传感器”设备类型中，FALSE=断开或无触发，TRUE=闭合或触发。

#### 6.2.7.5 事件

布尔状态功能集事件详见下表64。

表64 布尔状态功能集事件表

| 标识符  | 名称           | 重要等级 | 质量等级 | 通道 | 一致性 |
|------|--------------|------|------|----|-----|
| 0x00 | State Change | INFO | —    | V  | 0   |

其中，如果支持StateChange事件，则应在StateValue属性更改时生成。StateValue字段只准许表示StateValue属性的新值，详见下表65。

表65 StateValue 属性表

| 标识符 | 名称         | 类型   | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|------------|------|------|------|----|-----|
| 0   | StateValue | bool | —    | —    | —  | M   |

### 6.2.8 布尔状态配置功能集

#### 6.2.8.1 概述

该功能集用于配置布尔传感器，包括可选的状态变化报警功能以及与传感器相关的灵敏度级别配置。

#### 6.2.8.2 分类

布尔状态配置功能集分类详见下表66。

表66 布尔状态配置功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | BOOLCFG |

#### 6.2.8.3 Cluster ID

布尔状态配置功能集ID详见下表67。

表67 布尔状态配置功能集 ID 表

| 标识符    | 名称                        |
|--------|---------------------------|
| 0x0080 | BooleanStateConfiguration |

#### 6.2.8.4 特性

该功能集应支持FeatureMap位图属性，详见下表68。

表68 FeatureMap 位图属性表

| Bit | 编码      | 特征               | 一致性       | 概述         |
|-----|---------|------------------|-----------|------------|
| 0   | VIS     | Visual           | 0         | 支持视觉报警     |
| 1   | AUD     | Audible          | 0         | 支持听觉报警     |
| 2   | SPRS    | AlarmSuppress    | [VIS AUD] | 支持抑制或确认警报  |
| 3   | SENSLVL | SensitivityLevel | 0         | 支持设置传感器灵敏度 |

其中，AlarmSuppress功能应表明，当用户确认报警时，设备能够抑制支持的报警模式。这是为了在用户已经意识到传感器被触发，但不再希望设备上的报警模式处于活动状态时，停止视觉和/或听觉报警，例如：

用户已解决触发原因，但传感器尚未停止检测触发原因。

用户无法解决触发原因，但知道报警并在稍后抑制/确认报警。

对于该功能集的其余部分，对报警确认将被称为“抑制”。

被抑制的警报仍被视为处于活动状态，除非它被主动禁用或触发条件不再存在，否则将保持这种状态。抑制报警模式的动作仅适用于并旨在停止物理报警，例如发出声音或闪烁灯光；它不会影响AlarmsActive中的报警报告。

#### 6.2.8.5 数据类型

##### 6.2.8.5.1 AlarmModeBitmap 类型

此数据类型源自map8，详见下表69。

表69 AlarmModeBitmap 类型表

| Bit | 名称      | 概述   | 一致性 |
|-----|---------|------|-----|
| 0   | Visual  | 视觉报警 | VIS |
| 1   | Audible | 听觉报警 | AUD |

##### 6.2.8.5.2 SensorFaultBitmap 类型

此数据类型源自map16，详见下表70。

表70 SensorFaultBitmap 类型表

| Bit | 名称           | 概述        | 一致性 |
|-----|--------------|-----------|-----|
| 0   | GeneralFault | 检测到未指定的故障 | M   |

#### 6.2.8.6 属性

##### 6.2.8.6.1 属性列表

布尔状态配置功能集属性详见下表71。

表71 布尔状态配置功能集属性表

| 标识符    | 名称                         | 类型                | 约束条件                                  | 质量等级 | 默认 | 通道   | 一致性       |
|--------|----------------------------|-------------------|---------------------------------------|------|----|------|-----------|
| 0x0000 | CurrentSensitivityLevel    | uint8             | max<br>(SupportedSensitivityLevels-1) | N    | MS | RWVO | SENSLVL   |
| 0x0001 | SupportedSensitivityLevels | uint8             | 2to10                                 | F    | MS | RV   | SENSLVL   |
| 0x0002 | DefaultSensitivityLevel    | uint8             | max<br>(SupportedSensitivityLevels-1) | F    | MS | RV   | [SENSLVL] |
| 0x0003 | AlarmsActive               | AlarmModeBitmap   | all                                   | -    | 0  | RV   | VIS AUD   |
| 0x0004 | AlarmsSuppressed           | AlarmModeBitmap   | all                                   | -    | 0  | RV   | SPRS      |
| 0x0005 | AlarmsEnabled              | AlarmModeBitmap   | all                                   | N    | MS | RV   | [VIS AUD] |
| 0x0006 | AlarmsSupported            | AlarmModeBitmap   | all                                   | F    | 0  | RV   | VIS AUD   |
| 0x0007 | SensorFault                | SensorFaultBitmap | all                                   | -    | 0  | RV   | 0         |

#### 6.2.8.6.2 CurrentSensitivityLevel 属性

该属性应表明当前选择的灵敏度级别。

如果对该属性的写入交互包含不受支持的灵敏度值，则应返回CONSTRAINT\_ERROR状态。

#### 6.2.8.6.3 SupportedSensitivityLevels 属性

该属性应表明设备支持的灵敏度级别数量。

这些支持的灵敏度级别应按灵敏度排序，其中0值应被视为最低灵敏度级别（最不灵敏），而最大值应视为最高灵敏度级别。

所支持的灵敏度级别数量应代表设备支持的唯一灵敏度级别。

#### 6.2.8.6.4 DefaultSensitivityLevel 属性

该属性应表明制造商选择的默认灵敏度级别。

#### 6.2.8.6.5 AlarmsActive 属性

该属性应表明服务端上当前处于活动状态的特定报警模式。当传感器不再被触发时，对于所有支持的报警模式，应通过比特位设置为0，将此属性设置为不活动状态。

如果不支持该报警模式，则表明该报警模式的比特位应始终为0。一个比特位应该表明报警模式是否处于不活动状态：

0=不活动

1=活动

#### 6.2.8.6.6 AlarmsSuppressed 属性

该属性应表明服务端上当前抑制的特定报警模式。当传感器不再被触发时，对于所有支持的报警模式，应通过将比特位设置为0，将此属性设置为未抑制状态。

如果不支持该报警模式，则表明该报警模式的比特位应始终为0。一个比特位应该表明报警模式是否被抑制：

0=未抑制

1=抑制

#### 6.2.8.6.7 AlarmsEnabled 属性

该属性应表明传感器触发时将发出的报警模式。

如果不支持报警模式，则表明此报警模式的比特位应始终为0。一个比特位应该表明报警模式是启用还是禁用：

0=禁用

1=启用

6.2.8.6.8 AlarmsSupported 属性

该属性应表明传感器支持的报警。  
一个比特位应该表明是否支持报警模式：  
0=不支持  
1=支持

6.2.8.6.9 SensorFault 属性

该属性应表明设备记录的任何故障。

6.2.8.7 命令

6.2.8.7.1 布尔状态配置功能集命令

该功能集命令详见下表72。

表72 布尔状态配置功能集命令表

| 标识符  | 名称                 | 方向            | 返回 | 通道 | 一致性     |
|------|--------------------|---------------|----|----|---------|
| 0x00 | SuppressAlarm      | client→server | Y  | 0  | SPRS    |
| 0x01 | EnableDisableAlarm | client→server | Y  | 0  | VIS AUD |

6.2.8.7.2 SuppressAlarm 命令

SuppressAlarm命令详见下表73。

表73 SuppressAlarm 命令表

| 标识符 | 名称               | 类型              | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|------------------|-----------------|------|------|----|-----|
| 0   | AlarmsToSuppress | AlarmModeBitmap | all  | —    | —  | M   |

其中，AlarmsToSuppress字段应表明要抑制的报警模式。

如果不支持任何请求的报警模式，则应忽略此命令，服务端应返回CONSTRAINT\_ERROR状态码。

如果任何请求的报警模式未处于活动状态，例如传感器未被触发或报警模式被禁用/不支持，则应忽略此命令，服务端应返回INVALID\_in\_STATE状态码。

如果报警已被抑制，并且在AlarmsSuppressed属性中设置了特定比特位，则应忽略该比特位，并对剩余比特位进行评估。

对于AlarmsToSuppress字段中的有效比特位，设备应按照AlarmsToSuppress字段中的请求抑制指定的报警模式，并相应地设置AlarmsSuppressed属性。

6.2.8.7.3 EnableDisableAlarm 命令

EnableDisableAlarm命令详见下表74。

表74 EnableDisableAlarm 命令表

| 标识符 | 名称                    | 类型              | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------------|-----------------|------|------|----|-----|
| 0   | AlarmsToEnableDisable | AlarmModeBitmap | all  | —    | —  | M   |

其中，AlarmsToEnableDisable 字段应根据比特位状态表明启用或禁用的报警模式，如为AlarmsEnabled属性指定的。

如果不支持任何请求的报警模式，则应忽略此命令，服务端应返回CONSTRAINT\_ERROR状态码。

如果所有比特位都有效，则AlarmsEnabled属性的值应设置为AlarmsToEnableDisable字段的值。

如果启用了报警模式并且满足触发条件，设备应立即激活报警模式并在AlarmsActive属性中设置相关比特位。

如果禁用报警模式，则应清除处于活动或抑制状态的任何报警模式，并且应认为报警模式未处于活动状态。

6.2.8.8 事件

6.2.8.8.1 布尔状态配置功能集事件

布尔状态配置功能集事件详见下表75。

表75 布尔状态配置功能集事件表

| 标识符  | 名称                 | 重要等级 | 质量等级 | 通道 | 一致性     |
|------|--------------------|------|------|----|---------|
| 0x00 | AlarmsStateChanged | INFO | —    | V  | VIS AUD |
| 0x01 | SensorFault        | INFO | —    | V  | 0       |

6.2.8.8.2 AlarmsStateChanged 事件

该事件应在AlarmsActive和/或AlarmsSuppressed属性中的任何比特位更改后生成。这可能发生在以下情况下：当服务端的内部处理确定报警模式变为活动或不活动状态，或者处理SuppressAlarm或EnableDisableAlarm命令时，某些报警模式会变为抑制、活动或不活动。

如果多个报警模式同时改变状态，则允许发出一个合并了多个改变的单一事件，而不是每个改变代表一个单一事件的多个事件，详见下表76。

表76 AlarmsStateChanged 事件表

| 标识符 | 名称               | 类型              | 约束条件 | 质量等级 | 默认 | 一致性  |
|-----|------------------|-----------------|------|------|----|------|
| 0   | AlarmsActive     | AlarmModeBitmap | all  | —    | —  | M    |
| 1   | AlarmsSuppressed | AlarmModeBitmap | all  | —    | —  | SPRS |

——AlarmsActive 字段

当此事件发生时，该字段应表明报警模式的活动状态，如AlarmsActive属性。

——AlarmsSuppressed 字段

当此事件发生时，该字段应表明报警模式的抑制状态，如AlarmsSuppressed属性。

6.2.8.8.3 SensorFault 事件

当设备记录或清除故障时，应生成此事件，详见下表77。

表77 SensorFault 事件表

| 标识符 | 名称          | 类型                | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-------------|-------------------|------|------|----|-----|
| 0   | SensorFault | SensorFaultBitmap | all  | —    | —  | M   |

其中，当此事件发生时，SensorFault字段应表明SensorFault属性的值。

6.2.9 开关交互功能集

6.2.9.1 概述

这个功能集展示了与开关类设备的交互，以便其他设备使用这些交互。

支持两种类型的开关设备：锁定开关（例如摇杆开关）和瞬时开关（例如按钮），通过它们的功能标志进行区分。

与开关设备的交互被暴露为属性（对于锁定开关）和事件（对于两种类型的开关）。

感兴趣的客户端可以订阅这些属性/事件，从而了解交互情况，并可以根据此执行操作，例如通过发送命令来执行诸如控制灯或窗帘之类的操作。

6.2.9.2 分类

开关交互功能集的分类，详见下表78。

表78 开关交互功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | SWTCH   |

### 6.2.9.3 Cluster ID

开关交互功能集的ID，详见下表79。

表79 开关交互功能集 ID

| 标识符    | 名称     |
|--------|--------|
| 0x003B | Switch |

### 6.2.9.4 特性

该功能集的特性标志应该用于表明由功能集表示的设备的物理特性以及检测/报告能力，详见下表80。

表80 特性

| Bit | 编码  | 特征                        | 一致性           | 概述                                            |
|-----|-----|---------------------------|---------------|-----------------------------------------------|
| 0   | LS  | LatchingSwitch            | 0. a+         | 这是一个锁定开关设备                                    |
| 1   | MS  | MomentarySwitch           | 0. a+         | 这是一个瞬时开关设备                                    |
| 2   | MSR | MomentarySwitchRelease    | [MS&!AS]      | 开关支持释放事件生成                                    |
| 3   | MSL | MomentarySwitchLongPress  | [MS&(MSR AS)] | 开关支持长按检测                                      |
| 4   | MSM | MomentarySwitchMultiPress | AS, [MS&MSR]  | 开关支持多次按动检测                                    |
| 5   | AS  | ActionSwitch              | [MS]          | 开关是瞬时的，针对特定的用户操作（重点关注多次按压以及可选的长按）进行了简化的事件生成方案 |

其中，

- LatchingSwitch功能标志适用于在按下（或转动）后保持其位置的开关。
- MomentarySwitch功能标志适用于在按下（或转动）后不保持其位置的开关。在释放后，它会回到空闲位置。
- MomentarySwitchRelease功能标志适用于一个瞬时开关，它可以区分和报告释放事件。
- MomentarySwitchLongPress功能标志适用于一个瞬时开关，它可以区分和报告长按和短按。
- MomentarySwitchMultiPress功能标志适用于一个瞬时开关，它可以区分和报告双击和可能的多次按下，例如三次按下等。
- ActionSwitch功能标志表示简化了多按开关事件的处理。请参见章节“6.2.9.8 MultiPress事件的序列”。

### 6.2.9.5 属性

#### 6.2.9.5.1 属性列表

该功能集的属性，详见下表81。

表81 开关交互功能集属性表

| 标识符    | 名称                | 类型    | 约束条件                   | 质量等级 | 默认 | 通道 | 一致性 |
|--------|-------------------|-------|------------------------|------|----|----|-----|
| 0x0000 | NumberOfPositions | uint8 | min2                   | F    | 2  | RV | M   |
| 0x0001 | CurrentPosition   | uint8 | maxNumberOfPositions-1 | N    | 0  | RV | M   |
| 0x0002 | MultiPressMax     | uint8 | min2                   | F    | 2  | RV | MSM |

#### 6.2.9.5.2 NumberOfPositions 属性

这个属性应该指示开关具有的最大位置数。任何类型的开关都至少有2个位置。也可以参考章节“6.3.9.8 MultiPress事件的序列”，以获取NumberOfPositions>2的情况。

### 6.2.9.5.3 CurrentPosition 属性

这个属性应该指示开关的位置。有效范围是0到NumberOfPositions-1。CurrentPosition值0应该分配给开关的默认位置：例如摇杆开关的“open”状态，或按钮开关的“idle”状态。

### 6.2.9.5.4 MultiPressMax 属性

此属性应指明支持多次按压（设置了MSM功能标志）的瞬时开关可以检测并报告多少连续按压。

例如：一个瞬时开关支持单击、双击和三击，但不支持四击及以上，则会返回值3。

当在多次按压序列中检测到按压次数超过MultiPressMax时：

- 对于功能集修订版本小于2的服务端，应该生成一个MultiPressComplete事件，并将TotalNumberOfPressesCounted字段设置为MultiPressMax属性的值，且避免在按压序列恢复空闲之前（即不再处于多次按压计数状态）生成任何新的InitialPress或MultiPressOngoing事件。
- 对于功能集修订版本大于等于2的服务端，必须生成一个MultiPressComplete事件，并将TotalNumberOfPressesCounted字段设置为零（表示序列被中止），并且在按压序列恢复空闲之前不得生成任何新的InitialPress或MultiPressOngoing事件。

这样可以避免在有超出MultiPressMax的长按序列时，意外触发可能由开关专门处理的中间动作（例如触发特殊行为如恢复出厂设置，此类事件不应该向客户端生成）。

### 6.2.9.6 事件

#### 6.2.9.6.1 事件列表

该功能集的事件，详见下表82。

表82 开关交互功能集事件表

| 标识符  | 名称                 | 重要等级 | 通道 | 一致性     |
|------|--------------------|------|----|---------|
| 0x00 | SwitchLatched      | INFO | V  | LS      |
| 0x01 | InitialPress       | INFO | V  | MS      |
| 0x02 | LongPress          | INFO | V  | MSL     |
| 0x03 | ShortRelease       | INFO | V  | MSR     |
| 0x04 | LongRelease        | INFO | V  | MSL     |
| 0x05 | MultiPressOngoing  | INFO | V  | MSM&!AS |
| 0x06 | MultiPressComplete | INFO | V  | MSM     |

#### 6.2.9.6.2 SwitchLatched 事件

这个事件应该在锁定开关移动到新位置时生成。可能由于开关内的去抖动导致延迟。这个事件的数据应该包含以下信息，详见下表83。

表83 SwitchLatched 事件表

| 标识符 | 名称          | 类型    | 约束条件                   | 质量等级 | 默认 | 一致性 |
|-----|-------------|-------|------------------------|------|----|-----|
| 0   | NewPosition | unit8 | 0toNumberOfPositions-1 | -    | -  | M   |

其中，NewPosition字段应表明CurrentPosition属性的新值，例如在拨动开关后。

#### 6.2.9.6.3 InitialPress 事件

InitialPress事件应该在瞬时开关开始按下（去抖动后）时生成。这个事件的数据应该包含以下信息，详见下表84。

表84 InitialPress 事件表

| 标识符 | 名称          | 类型    | 约束条件                   | 质量等级 | 默认 | 一致性 |
|-----|-------------|-------|------------------------|------|----|-----|
| 0   | NewPosition | unit8 | 0toNumberOfPositions-1 | -    | -  | M   |

其中，NewPosition字段应该表明CurrentPosition属性的新值，例如在按下按钮后。

6.2.9.6.4 LongPress 事件

这个事件应当在瞬时开关被按下“较长”时间后生成。判定“较长”时间的阈值由制造商决定，因为这取决于开关的物理特性。

- 当设置了AS功能标志时，该事件：
  - 不应在多次按压序列中生成（因为长按是与多次按压完全分离的周期）；
  - 只应当在多次按压完成（MultiPressComplete）后第一次InitialPress并在空闲期后检测到长按时才会生成。
- 否则，当设置了MSM功能标志时，该事件：
  - 不应在多次按压序列中生成（因为长按是与多次按压完全分离的周期）；
  - 只应当在多次按压完成（MultiPressComplete）后第一次InitialPress并在空闲期后检测到长按时才会生成；
  - 不应在没有中间MultiPressComplete事件的情况下，在MultiPressOngoing事件之后生成。

以上限制意味着，对于一个同时设置了MSM和/或MSL功能标志的开关而言，某次检测到的完整交互过程要么是一次单独的长按检测周期（包括InitialPress、LongPress、LongRelease），要么是一次多次按压检测周期（以MultiPressComplete结束）。本来会被报告为长按的按压，在多次按压检测过程中会转换成计数到MultiPressComplete事件中的一次按压，以及（在适用的情况下）对应的InitialPress/ShortRelease组合。

做此限制的原因在于，当把长按和多次按压事件混合使用时，事件的解释会陷入歧义。这个事件的数据应该包含以下信息，详见下表85。

表85 LongPress 事件表

| 标识符 | 名称          | 类型    | 约束条件                   | 质量等级 | 默认 | 一致性 |
|-----|-------------|-------|------------------------|------|----|-----|
| 0   | NewPosition | unit8 | 0toNumberOfPositions-1 | -    | -  | M   |

其中，NewPosition字段应该表明CurrentPosition属性的新值，例如在按下按钮后。

6.2.9.6.5 ShortRelease 事件

如果服务端设置了ActionSwitch（AS）功能标志，则根本不应生成此事件，因为设置ActionSwitch功能标志会禁止设置瞬时开关短释放（MomentarySwitchShortRelease，MSR）功能标志。否则，以下段落将描述生成此事件的情况。

当瞬时按钮被释放（去抖动后）时，应该生成此事件。

- 如果服务端设置了MomentarySwitchLongPress（MSL）功能标志，那么当开关被释放时，如果自上一次InitialPress事件以来没有生成过LongPress事件，则应生成此事件。
- 如果服务端未设置MomentarySwitchLongPress（MSL）功能标志，那么在开关被释放时应生成此事件——即使开关被长时间按下。

可以同时参见章节“6.2.9.7 生成事件的顺序”。这个事件的数据应该包含以下信息，详见下表86。

表86 ShortRelease 事件表

| 标识符 | 名称               | 类型    | 约束条件                   | 质量等级 | 默认 | 一致性 |
|-----|------------------|-------|------------------------|------|----|-----|
| 0   | PreviousPosition | unit8 | 0toNumberOfPositions-1 | -    | -  | M   |

其中，PreviousPosition字段应该表明CurrentPosition属性的旧值，即在释放之前。

### 6.2.9.6.6 LongRelease 事件

当瞬时开关被释放（去抖动后）并且在长时间按下后，应该生成此事件，即在释放之前，如果自上次InitialPress事件以来已生成LongPress事件，则应该生成此事件。也可以参见“6.2.9.7 生成事件的顺序”。

这个事件的数据应该包含以下信息，详见下表87。

表87 LongRelease 事件表

| 标识符 | 名称               | 类型    | 约束条件                   | 质量等级 | 默认 | 一致性 |
|-----|------------------|-------|------------------------|------|----|-----|
| 0   | PreviousPosition | unit8 | 0toNumberOfPositions-1 | -    | -  | M   |

其中PreviousPosition字段应该表明CurrentPosition属性的旧值，即在释放之前。

### 6.2.9.6.7 MultiPressOngoing 事件

如果服务端设置了动作开关（AS）功能标志，则根本不应生成此事件。否则，以下段落将描述生成此事件的情况。

这个事件应该生成，以指示在多次按下的序列中瞬时开关被按下了多少次。参见“6.2.9.8 MultiPress事件的序列”详细信息。详见下表88。

表88 MultiPresssOngoing 事件表

| 标识符 | 名称                          | 类型    | 约束条件                  | 质量等级 | 默认 | 一致性 |
|-----|-----------------------------|-------|-----------------------|------|----|-----|
| 0   | NewPosition                 | unit8 | 0toNumberOfPosition-1 | -    | -  | M   |
| 1   | CurrentNumberOfPressCounted | unit8 | 2toMultiPressMax      | -    | -  | M   |

其中，

- NewPosition字段应该表明CurrentPosition属性的新值，例如在按下按钮后。
- CurrentNumberOfPressCounted字段应该包含：
  - a) 当检测到多次按下序列的第二次按下时，值为 2；
  - b) 当检测到多次按下序列的第三次按下时，值为 3；
  - c) 当检测到多次按下序列的第 N 次按下时，值为 N。

### 6.2.9.6.8 MultiPressComplete 事件

这个事件应该生成，以指示在多次按下的序列中瞬时开关被按下了多少次。参见“6.2.9.8 MultiPress事件的序列”详细信息。

这个事件的数据应该包含以下信息，详见下表89。

表89 MultiPressComplete 事件表

| 标识符 | 名称                          | 类型    | 约束条件                  | 质量等级 | 默认 | 一致性 |
|-----|-----------------------------|-------|-----------------------|------|----|-----|
| 0   | PreviousPosition            | unit8 | 0toNumberOfPosition-1 | -    | -  | M   |
| 1   | TotalNumberOfPressesCounted | unit8 | maxMultiPressMax      | -    | -  | M   |

PreviousPosition字段应该表明CurrentPosition属性的旧值，即在释放之前。

TotalNumberOfPressesCounted字段应该包含：

- 当多次按下序列中止，按压次数超过MultiPressMax时，值为0，
- 当多次按下序列中只按下1次时（并且序列已经结束），即没有双按（或更多）时，值为1，
- 当多次按下序列中有两次按下时（并且序列已经结束），值为2，
- 当多次按下序列中有三次按下时（并且序列已经结束），值为3，
- 当多次按下序列中有N次按下时（并且序列已经结束），值为N。

注：TotalNumberOfPressesCounted支持值0的引入可能会影响使用功能集修订版本1的开关的客户端，因为此类服务端不会使用TotalNumberOfPressesCounted的这个值来指示中止的序列。客户端应始终考虑TotalNumberOfPressesCounted字段进行操作，因为对于从1到MultiPressMax的值，导致事件发生的用户操作会因计数不同而不同。

## 6.2.9.7 生成事件的顺序

### 6.2.9.7.1 概述

这个部分描述了三种类型的瞬时开关（通过它们的功能标志进行区分）将生成的事件序列。对于每种开关类型，将定义以下三种交互的生成事件序列：

1. 按下瞬时开关的序列。
2. 长时间按下开关的序列
3. 非常长时间按下开关的序列

在下面的子节中描述的三种交互中，如果NumberOfPositions属性等于2，则InitialPress和LongPress事件的新Position字段设置为1，ShortRelease和LongRelease事件的PreviousPosition字段设置为1。对于NumberOfPositions属性的较大值，请参见章节“6.2.9.10 MultiPosition详细信息”。

### 6.2.9.7.2 支持 long-press 检测的开关（设置 MSL 功能标志）

这个开关（同时设置MS&MSL标志的功能）应该生成任意一个交互周期的两个或三个（取决于开关按下的时间长短）事件序列，顺序如下所示，并在下图8中说明。

- 短按时：
  - 生成InitialPress事件
  - 如果设置了MSR功能标志且未设置AS功能标志，则在检测到释放时，在前一个事件之后生成ShortRelease事件。
  - 如果设置了MSM功能标志，则在前一个事件之后生成MultiPressComplete(1)事件。
- 长按（或非常长按）时：
  - 生成InitialPress事件
  - 在前一个事件之后检测到长按时生成LongPress事件
  - 在前一个事件之后检测到开关释放时生成LongRelease事件（而不是MultiPressComplete(1)事件）。

在这种情况下，LongPress事件应该仅生成一次，并且不应该重复，无论开关按下多长时间。

该图像显示了开关状态（低=未按下，高=按下）的时间表示，其中彩色点表示在该时刻生成的各种事件。

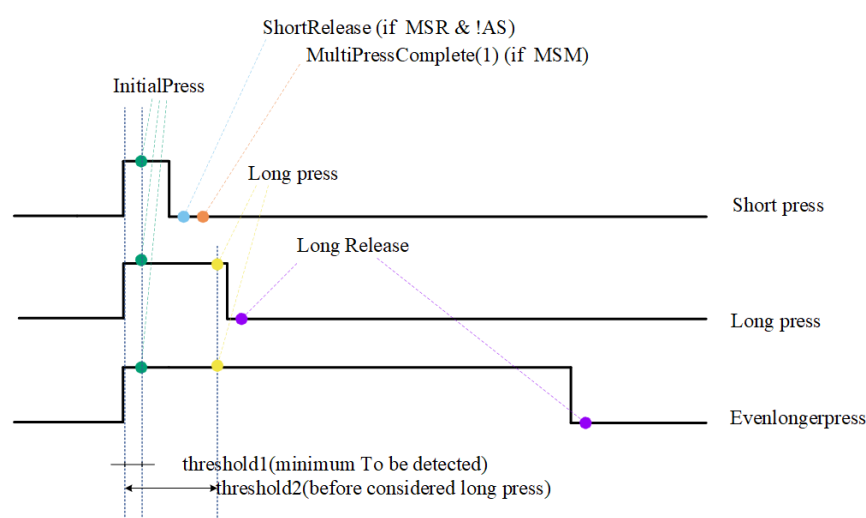


图8 已设置 MSL 功能标志的开关设备（单次按下情况）

### 6.2.9.7.3 不支持 long-press 检测的开关（未设置 MSL 功能标志）

该开关（具有MS&!MSL标志的功能）不应该生成LongPress和LongRelease事件，因此它应该生成一个交互周期的两个事件序列，无论开关按下多长时间，顺序如下所示，并在下图9中说明，基于是否设置了MSM、AS和MSR标志。

任何按压长度：

- 1) 生成 InitialPress 事件
- 2) 如果设置了 MSR 功能标志且未设置 AS 功能标志，则在检测到释放时，在前一个事件之后生成 ShortRelease 事件。
- 3) 如果设置了 MSM 功能标志，则在前一个事件之后生成 MultiPressComplete (1) 事件。

注：即使“长时间”按下后，释放事件也不会是LongRelease。没有MSL功能标志的设备不应生成LongPress和LongRelease事件。

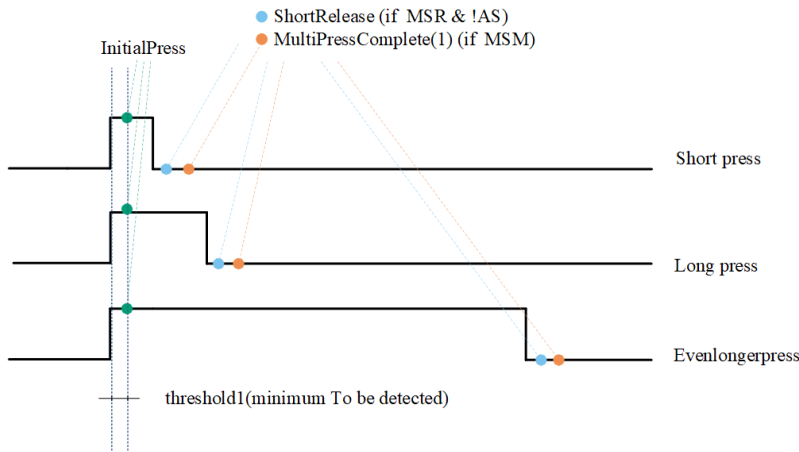


图9 未设置MSL功能标志的开关设备（单次按下情况）

6.2.9.7.4 已设置最基本MS-only功能标志的开关，不包括（MSR|MSL|MSM|AS）

如下图10所示，该开关（功能标志状态为MS&!MSR&!MSL&!MSM&!AS）应该在一个交互周期生成一个InitialPress事件，无论开关按下多长时间。

一个拥有此功能标志集的设备不应该生成任何ShortRelease、LongPress和LongRelease事件。

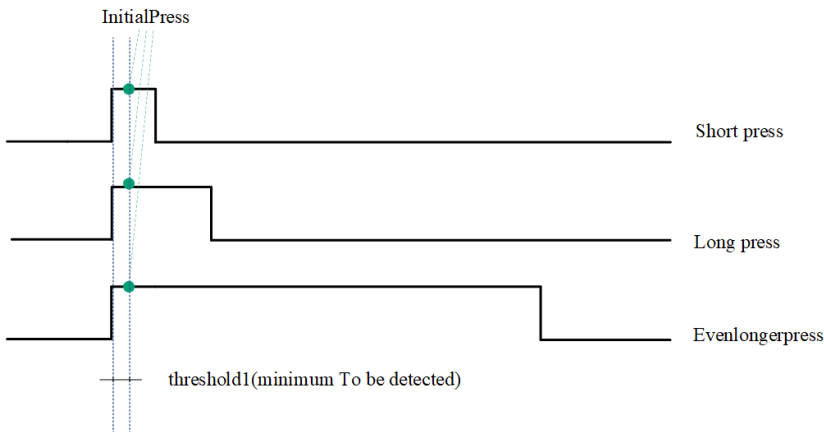


图10 开关设备只传送“InitialPress”事件（单次按压情况）

6.2.9.8 MultiPress事件的序列

6.2.9.8.1 概述

多次按下检测是瞬时开关的一个特性（用MSM功能标志表示，AS功能标志可选），它们可以在一定的时间范围内计数和报告按下-释放周期的序列，例如，指示用户已经按下开关一次、两次或三次（甚

至更多)。这种检测的时间窗口的定义是制造商特定的,因为它取决于开关物理。可以检测和报告的最大按下次数应该在MultiPressMax属性中指示。MultiPressMax的最小值和默认值都是2。

支持MultiPress的开关应该设置功能标志MS&MSM,并且可以根据每个标志的符合性规则可选地设置MSL、MSR和AS标志。

ActionSwitch (AS) 功能标志是一个行为选择器,设置时会减少开关生成的事件数量。

当设置了MSM时,按压周期要么是长按周期(如果设置了MSL),要么是多次按压周期。

- 如果设置了MSL功能标志,并且在InitialPress之后的第一个按压周期被认为是“长按”,那么开关应按照章节“支持long-press检测的开关(设置MSL功能标志)”中描述的方式做出反应,并且不应将其视为多次按压序列的开始(参见“图11”中的倒数第二个案例)。
- 如果未设置MSL功能标志,或者第一个按压周期较短,那么开关应计数按压次数,直到开关变得足够空闲,并且不应为多次按压周期中的中间长按生成LongPress和LongRelease事件。

#### 6.2.9.8.2 未设置操作切换(AS)功能标志时的Multi-Press 行为

具有MSM功能标志但未设置AS标志的开关:

- 在按钮多次按压周期内的每次按钮按下开始时,必须生成InitialPress事件;
- 如果设置了MSR,则在按钮多次按压周期内的每次按钮按下结束时,必须生成ShortRelease事件;
- 在按钮多次按压周期的第二次及后续按压的InitialPress事件之后,必须生成MultiPressOngoing事件;
- 在多次按压周期结束时,必须生成MultiPressComplete事件,包括单次按压;
- 在多次按压周期内,不得(对于功能集修订版本 $\geq 2$ )或不应(对于功能集修订版本 $< 2$ )生成任何LongPress和LongRelease事件。

MultiPressOngoing事件是为了让感兴趣的各方来跟踪多次按下序列中的实际按下。MultiPressComplete事件是为了让感兴趣的各方了解整个序列的结果:在多次按下序列结束后,他们将收到MultiPressComplete事件,指示开关被按下的次数。

在下图中,指示了若干个用户交互序列:

- 单次按下序列:在按下和释放时刻之后,应该生成InitialPress,如果MSR功能标志已设成MultiPressComplete(1),因为它检测到序列由一次按下组成。对于这种情况,不应该生成MultiPressOngoing事件。
- 双击按下序列:在每次按下和释放时刻之后,应该生成InitialPress事件,如果MSR功能标志已设置,则应生成ShortRelease事件。此外,当开关第二次按下时,应该生成MultiPressOngoing(2)事件,因为开关检测到第二次按下。请注意,此事件与第二个InitialPress事件重合;两者都应该生成。在一段时间后,当开关检测到没有第三次按下时,它应该生成MultiPressComplete(2),因为它检测到序列由两次按下组成。
- 第三次按下序列:在每次按下和释放时刻之后,应该生成InitialPress事件,如果MSR功能标志已设置,则应生成ShortRelease事件。此外,当开关第二次按下时,应该生成MultiPressOngoing(2)事件,因为开关检测到第二次按下。请注意,此事件与第二个InitialPress事件重合;两者都应该生成。此外,当开关第三次按下时,应该生成MultiPressOngoing(3)事件,因为开关检测到第三次按下。请注意,此事件与第三个InitialPress事件重合;两者都应该生成。在一段时间后,当开关检测到没有第四次按下时,它应该生成MultiPressComplete(3),因为它检测到序列由三次按下组成。
- 这种情况下,特定地不生成LongPress和LongRelease事件(尽管下图中在某些情况下第二次按下的持续时间较长),因为多次按下序列不应/不得在序列中生成LongPress和LongRelease事件(更多详细信息请参见章节“6.2.9.6.4 LongPress事件”)。
- 以长按开始的N次按下(如果设置了MSL功能标志):这从一个长按序列开始,应按章节“支持long-press检测的开关(设置MSL功能标志)”进行考虑。因此它是一个类似于三次按下的序列,只是按下次数为N而不是3。

对于需要同时生成多个事件的上述情况,MultiPressOngoing事件应该在InitialPress事件之后直接生成。

注1：图11中括号内的数字表示事件数据中CurrentNumberOfPressesCountedresp. TotalNumberOfPressesCounted字段的值。

注2：与其他图表一样，需要考虑足够的去抖动时间来检测按下和释放事件。这一点已包含在图11中，但为了便于阅读，在上文的表述中没有提及，但必须应用。

注3：在最终系统中，几个事件需要非常低的拥塞和延迟才能用于实时控制和使用案例。在设计过程中，必须注意，如果低延迟的假设不成立，则不要依赖于非常低的延迟和实时事件时间戳。

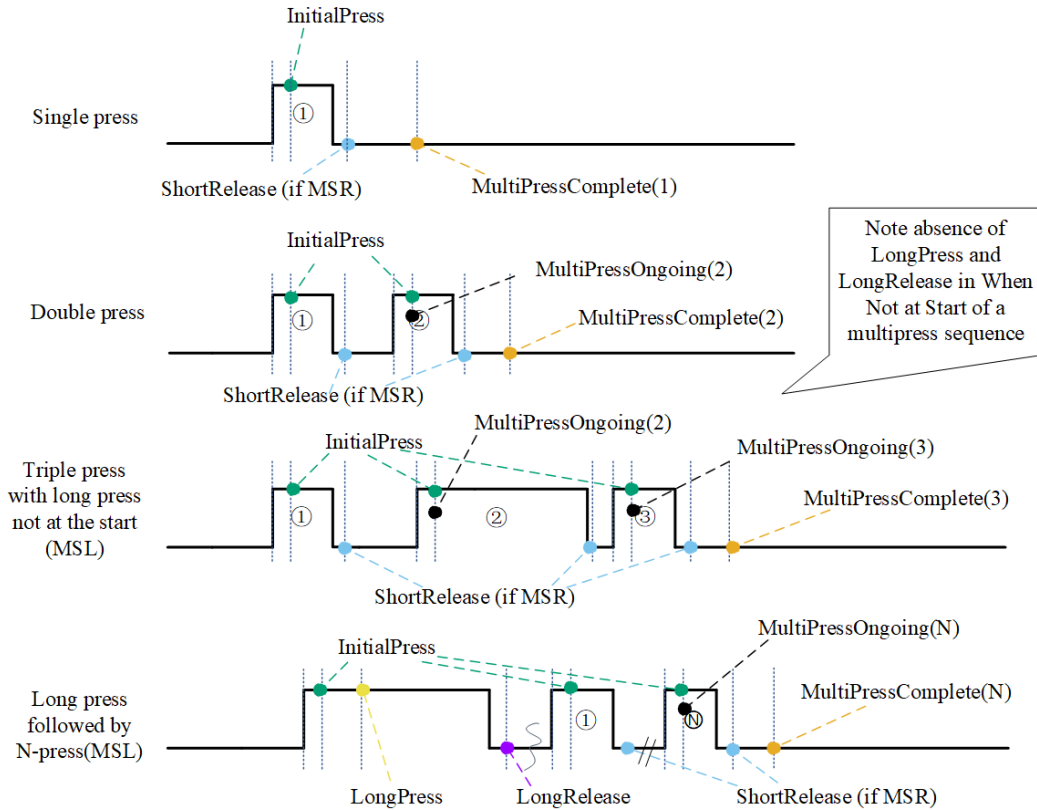


图11 带 MSM 和 !AS 功能标志的开关设备（多按情况）

#### 6.2.9.8.3 设置操作切换（AS）功能标志时的多按行为

当设置了ActionSwitch（AS）功能标志时，所有按压的行为如下：

- 开关应仅在每个按钮按压周期开始时生成InitialPress事件（无论是否长按）；
- 开关不应生成ShortRelease事件；
- 开关不应生成MultiPressOngoing事件。

由于这些简化，客户端可以仅根据以下内容来确定用户操作：

- LongRelease（如果设置了MSL功能标志）用于确定长按的结束。
- 如果需要，InitialPress和LongPress（如果设置了MSL功能标志），用于检测按压和长按周期的开始，以便支持诸如“长按时执行动作X并在释放时停止”的操作。
- MultiPressComplete用于确定一系列按压的结束（0..MultiPressMax）
- 仅提供开关检测到的最终按压次数。
- 值为0表示按压中止（例如，可能连续按压次数过多，按钮按下时间过长等）。
- 值为1表示单次按压，2表示双击，依此类推。

由于省略了ShortRelease、MultiPressOngoing和inner InitialPress事件，生成了更少的网络流量，同时不会遗漏最终用户的整体“操作”意图。

在下图12中，指示了若干个用户交互序列：

- 单次按下序列：在初始按下检测后，生成InitialPress事件。经过一段时间后，当开关检测到没有第二次按压，并且按压不是长按（如果设置了MSL功能标志），生成MultiPressComplete（1）事件，因为它检测到序列由一次按压组成。

- 双击按下序列：在初始按下检测后，生成InitialPress事件。经过一段时间后，当开关检测到没有第三次按下时，生成MultiPressComplete (2) 事件，因为它检测到序列由两次按压组成。
- N次按下序列：在初始按下检测后，生成InitialPress事件。此外，当开关第二次按下时，我们注意到按下时间较长，但没有生成LongPress或LongRelease事件，因为多次按压序列中的长按（不是在开始时）被视为短按。经过一段时间后，当开关检测到没有超过N次按下时，它生成MultiPressComplete (N) 事件，因为它检测到序列由N次按下组成。
- 长按序列后跟N次按下序列：
  - 这种情况包含两个独立的序列，因为长按序列始终是一个独立的序列，并且不会在多次按下中报告为“长按”（参见图12示例情况）。
  - 第一个序列是一个长按序列，包括：InitialPress、LongPress，最后是LongRelease。
  - 第二个序列，无论是否立即发生，都是一个N次按下序列，包括：InitialPress，最后是MultiPressComplete (N) 。
- N次按下序列，其中 $N > \text{MultiPressMax}$ ：在初始按下检测后，生成InitialPress事件。经过一段时间后，当开关检测到没有超过N次按压时，它生成MultiPressComplete (0) 事件，因为它检测到多次按压序列已完成，但由于按压次数超过了MultiPressMax，序列使用0作为TotalNumberOfPressesCounted来表示无效的完成序列。

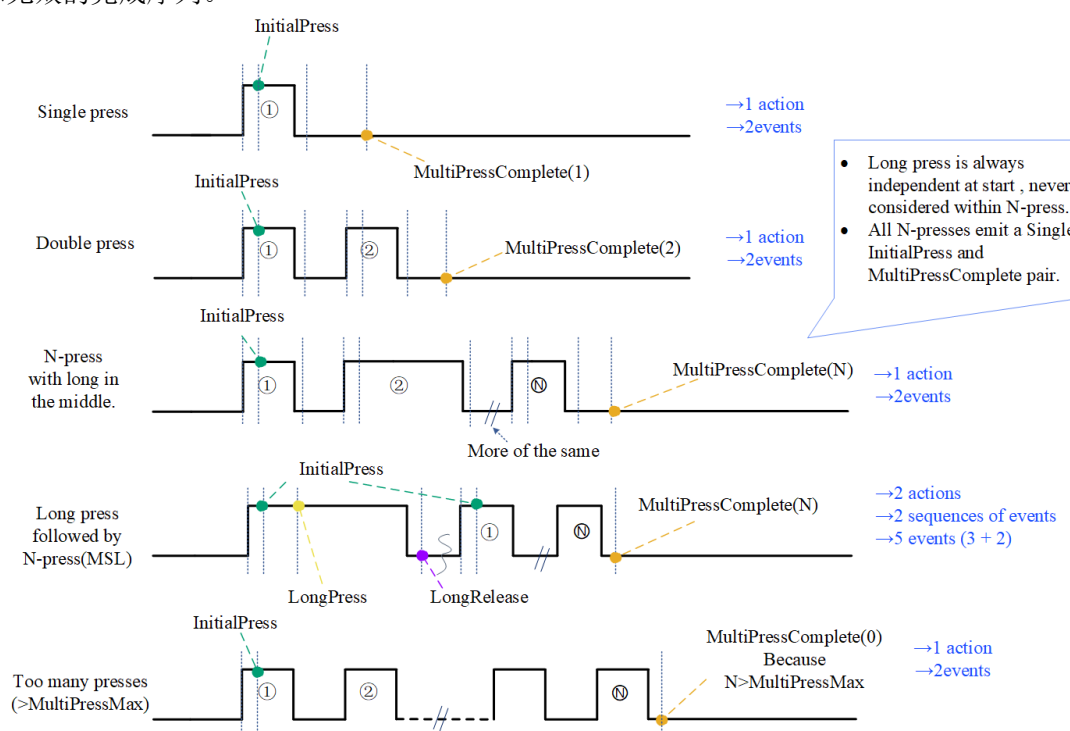


图12 带 MSM 和 !AS 功能标志的开关设备（多按情况）

#### 6.2.9.9 MS 功能标志的案例汇总

鉴于不同开关功能可能引发的各种情况，以下描述了可以明确传达用户操作意图的边界：

- MS&!MSR&!MSM：每个操作周期只有一个InitialPress事件。
- MS&!MSM：每个操作周期以ShortRelease或LongRelease（如果设置了MSL功能标志）结束，具体取决于长按或短按以及是否设置了MSR/MSL功能标志。
- MS&MSM：每个操作周期以LongRelease（如果设置了MSL功能标志且第一次按下时间较长）或MultiPressComplete结束。
  - AS功能标志不会改变哪些事件可用于动作检测的结果。
  - AS功能标志与此功能集的先前修订版本向后兼容。
  - AS功能标志主要提供减少生成的事件数量。

上述适用于客户端在操作过程中不使用指示操作周期进展的事件（即ShortRelease、MultiPressOngoing、重复的InitialPress）的情况。

#### 6.2.9.10 MultiPosition 详细信息

##### 6.2.9.10.1 概述

这个部分将会讨论一些拥有多于2个位置的开关设备的原型，以及它们应该如何设置属性值和生成与其特性相匹配的事件。

对于SwitchLatched、InitialPress、LongPress和MultiPressOngoing事件，NewPosition字段应该设置为对应于开关移动到的新位置的值。对于ShortRelease、LongRelease和MultiPressComplete事件，PreviousPosition字段应该设置为刚刚释放的开关的位置对应的值。

##### 6.2.9.10.2 具有N个稳定位置（ $N > 2$ ）和固定顺序的 LatchingSwitch

使用这样的设备，用户可以将开关从位置M移动到位置M-1和M+1（在端点位置之间进行环绕，或在端点位置处进行固定停止）。在与开关设备的每次交互中，它应该生成一个SwitchLatched事件，其中NewPosition字段设置为与新位置相关联的值。由于物理约束，这样的事件将具有NewPosition字段，该字段等于前一个NewPosition字段加或减1（如果开关没有端点，则为NumPositions的模）。

在第一个例子中，开关有3个位置，与值0、1和2相关联。在这种情况下，环绕是不可能的：从位置0它只能移动到位置1。如图13所示。

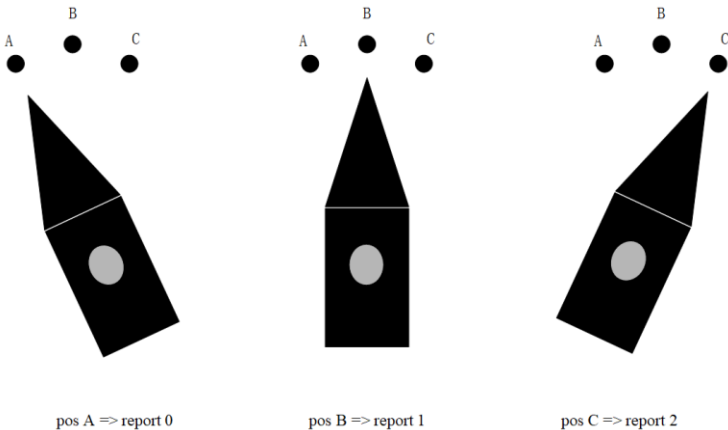


图13 三位置旋转开关设备

在另一个例子中，开关有8个位置，与值0到7相关联。在这种情况下，开关的物理特性允许环绕：从位置0可以移动到位置1或位置7。如图14所示。

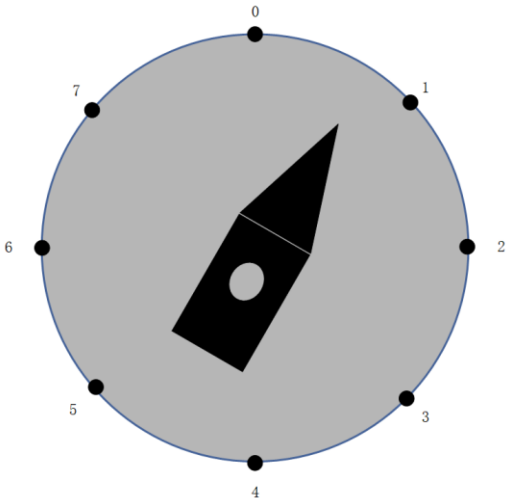


图14 具有 8 个位置 and 环绕功能的旋转开关装置

### 6.2.9.10.3 具有N个稳定位置 (N>2) 和随机顺序 (例如：收音机按钮) 的 LatchingSwitch

使用这个设备，用户可以按下任何可用的按钮，因此这个开关不显示我们讨论的具有固定顺序的锁定开关的增量或减量行为。在下图15所示的示例中，5个按钮用用户标记为“A”到“E”，并与值0到4相关联。用户首先按下“A”按钮，开关设备生成一个SwitchLatched事件，其中NewPosition设置为0。然后用户首先按下“D”按钮，开关设备生成一个SwitchLatched事件，其中NewPosition设置为4。如图15所示。

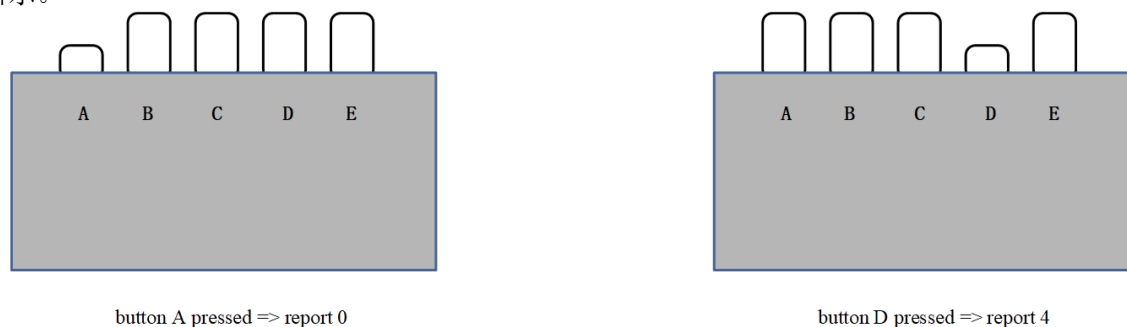


图15 具有单选按钮的开关设备

### 6.2.9.10.4 具有2个或多个非稳定位置的 MomentarySwitch

对于具有多个稳定位置的MomentarySwitch，这取决于开关设备的物理特性，将生成哪个事件序列。

注：在这一节中，我们只提到InitialPress和ShortRelease事件。开关设备还可以生成上面为瞬时开关定义的其他事件，这取决于开关设备的功能和与开关设备的交互。

第一个变体（下图16，示例：窗帘的上/下控制）显示了一个开关处于中性位置（图16的最左图），它对应于CurrentPosition=0。用户可以按下顶部（位置值1）或底部（位置值2）。不能直接从位置1转到位置2或反之亦然——开关总是需要通过中性位置0。

所以当用户按下开关的顶部时，将生成InitialPress (NewPosition=1) 事件。当他们释放顶部时，将生成 ShortRelease (PreviousPosition=1) 事件。用户继续按下底部，事件 InitialPress (NewPosition=2) 被生成。当底部释放时，事件ShortRelease (PreviousPosition=2) 被生成。

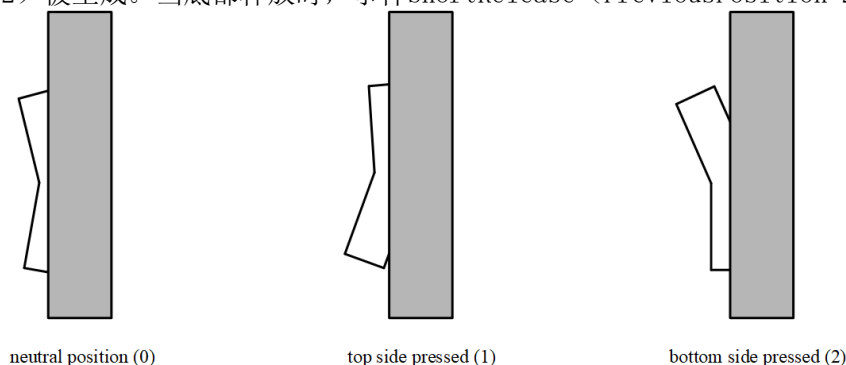


图16 上下控制开关设备

另一种变体（下图17，示例：操纵杆）具有一个中间的中性位置的控制手柄（图17的最左图），它对应于CurrentPosition=0。手柄可以沿着虚线移动。

在中间的图中，用户将手柄移动到东侧的位置，然后释放它（使其返回到中性中间位置）。这将生成以下事件序列：

```
InitialPress (NewPosition=3)    //移动到东侧位置
ShortRelease (PreviousPosition=3) //从东返回到中性中间位置
```

在右图中，用户将手柄移动到西南位置，然后移动到南位置，然后释放它（使其返回到中性中间位置）。这将生成以下事件序列：

```
InitialPress (NewPosition=6)    //移动到西南位置
InitialPress (NewPosition=5)    //移动到南侧位置
```

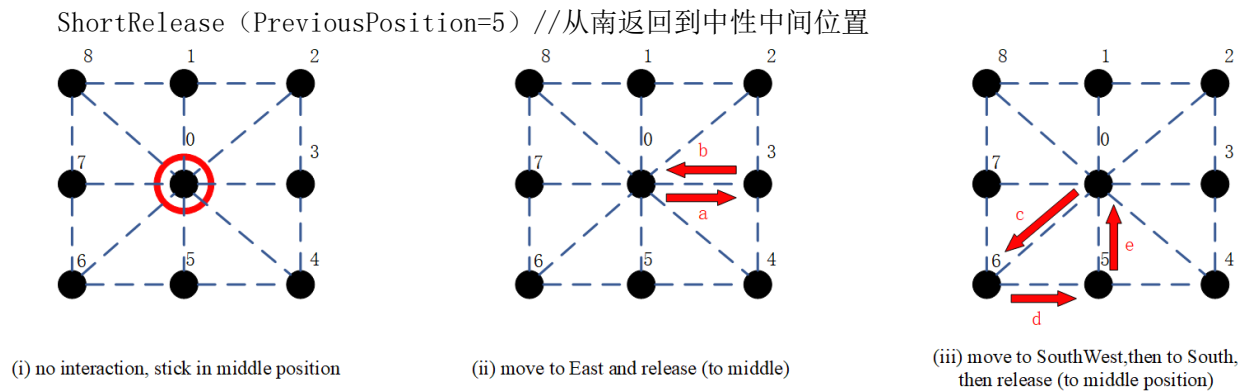


图17 带摇杆的开关设备

因此，在“操纵杆”变体中，可能会有一系列的InitialPress事件，而没有中间的ShortRelease事件——不像“窗帘控制”变体，它总是有这样一个中间的ShortRelease事件。

6.2.10 警报基础功能集

6.2.10.1 概述

该集群是一个基础集群，可以从中派生出特定设备类型的警报集群。每个派生集群都应该定义此集群中使用的AlarmMap数据类型的值。每个派生都应该定义此集群中使用的AlarmBitmap数据类型的值。每个派生集群都应该定义哪些警报是锁存的。每个派生都应该定义哪些警报是锁存的。

6.2.10.2 分类

该功能集的分类，详见下表90。

表90 警报基础功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | ALARM   |

6.2.10.3 ClusterID

警报基础功能集ID详见下表91。

表91 警报基础功能集 ID 表

| 标识符 | 名称        |
|-----|-----------|
| n/a | AlarmBase |

6.2.10.4 特性

该集群应支持以下定义的FeatureMap位图属性，详见下表92。

表92 警报基础功能集位图特性表

| Bit | 编码    | 特征    | 概述       |
|-----|-------|-------|----------|
| 0   | RESET | Reset | 支持重置警报功能 |

其中，重置特性，表示可以通过重置命令来重置警报。

6.2.10.5 数据类型

AlarmBitmap数据类型应该是一个map32，其值由派生集群定义。每个位位置的含义对于派生集群中的所有属性来说都是一致的。也就是说，如果位0为某个警报所定义，那么该警报的Latch（锁存）、State（状态）和Supported（支持）信息也都在位0上。

6.2.10.6 属性

该功能集的属性详见下表93。

表93 警报基础功能集属性表

| 标识符    | 名称        | 类型          | 约束条件 | 质量等级 | 默认 | 通道 | 一致性   |
|--------|-----------|-------------|------|------|----|----|-------|
| 0x0000 | Mask      | AlarmBitmap | all  |      | 0  | RV | M     |
| 0x0001 | Latch     | AlarmBitmap | all  | F    | 0  | RV | RESET |
| 0x0002 | State     | AlarmBitmap | all  |      | 0  | RV | M     |
| 0x0003 | Supported | AlarmBitmap | all  | F    | 0  | RV | M     |

其中，

- Mask属性应指示一个位图，其中Mask属性中设置的每个位对应于应启用的警报。
- Latch属性应指示一个位图，其中Latch属性中的每个设置的位应指示对应的警报在设置时将会被锁存，且当导致警报的基本条件不再存在时不会重置为非活动状态，因此需要使用重置命令进行显式重置。
- State属性应表示一个位图，其中每个位代表一个警报的状态。true的值表示警报是激活的，否则警报是不活动的。
- Supported属性应指示一个位图，其中每个位都表示一个警报是否被支持。true的值意味着支持该警报，否则该警报不被支持。如果一个警报不被支持，那么Mask、Latch和State中相应的位应该为false。

#### 6.2.10.7 命令

##### 6.2.10.7.1 警报基础功能集命令

该功能集的命令详见下表94。

表94 警报基础功能集命令表

| 标识符  | 名称                  | 方向            | 返回 | 通道 | 一致性   |
|------|---------------------|---------------|----|----|-------|
| 0x00 | Reset               | client→server | Y  | 0  | RESET |
| 0x01 | ModifyEnabledAlarms | client→server | Y  | 0  | 0     |

##### 6.2.10.7.2 重置命令

该命令会重置当前的和已锁定的警报（如果可能的话）。任何生成的通知事件都应包含字段，表示在命令处理后服务端的状态，详见下表95。

表95 重置命令表

| 标识符 | 名称     | 类型          | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|--------|-------------|------|------|----|-----|
| 0   | Alarms | AlarmBitmap | all  | -    | 0  | M   |

其中，Alarms字段应指示一个位图，其中该字段中设置的每个位对应于在状态属性中应重置为非活动状态的警报，除非警报定义要求手动干预。如果成功重置指示的警报，则响应状态代码应为成功，否则，响应状态代码应为失败。

##### 6.2.10.7.3 ModifyEnabledAlarms 命令

该命令允许客户端请求服务端启用或禁止警报，详见下表96。

表96 ModifyEnabledAlarms 命令表

| 标识符 | 名称   | 类型          | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|------|-------------|------|------|----|-----|
| 0   | Mask | AlarmBitmap | all  | -    | 0  | M   |

其中，Mask字段应指示一个位图，其中该字段中设置的每个位对应于应启用或抑制的警报。值为1应指示应启用警报，而值为0应指示应抑制警报。如果服务端接收到包含对未知警报设置的位的掩码的此命令，则应使用状态码“INVALID\_COMMAND”进行响应。

收到此命令的服务端，如果其掩码中包含已设置但不受支持的警报位，就应根据Supported属性中的指示，以“INVALID\_COMMAND”状态码进行响应。无法启用当前被抑制的警报，或无法抑制当前已启用的警报的服务端应该以FAILURE的状态码做出响应；否则，服务端应该以“SUCCESS”的状态码做出响应。

在“SUCCESS”的情况下，服务端还应将Mask属性的值更改为此命令的Mask字段的值。之后，服务端还应更新其State属性的值，以反映由Mask属性的新值指示的新警报集的状态。

### 6.2.10.8 事件

#### 6.2.10.8.1 警报基础功能集事件

该功能集事件详见下表97。

表97 警报基础功能集事件表

| 标识符  | 名称     | 重要等级 | 质量等级 | 通道 | 一致性 |
|------|--------|------|------|----|-----|
| 0x00 | Notify | INFO | —    | V  | M   |

#### 6.2.10.8.2 Notify 事件

该事件将在一个或多个警报状态发生变化时生成，并应具有以下字段，详见下表98。

表98 Notify 事件表

| 标识符 | 名称       | 类型          | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|----------|-------------|------|------|----|-----|
| 1   | Active   | AlarmBitmap | all  | —    | 0  | M   |
| 2   | Inactive | AlarmBitmap | all  | —    | 0  | M   |
| 3   | State    | AlarmBitmap | all  | —    | 0  | M   |
| 4   | Mask     | AlarmBitmap | all  | —    | 0  | M   |

其中，

- Active字段应该指示那些已经变成激活状态的警报。
- Inactive字段应该指示那些已经变成未激活状态的警报。
- Mask字段应为此事件生成时Mask属性的副本。
- State字段应该是生成事件时产生的新State属性值的副本。换句话说，该字段应该包含所有Active集中设置的位，并且不应该包含任何Inactive集中设置的位。

### 6.3 测量与传感

#### 6.3.1 概述

本文档中所规定的功能集是通用的测量和传感接口，其通用性很强，足以在广泛的应用领域中发挥作用。

##### 6.3.1.1 Cluster ID

下表99列出了本章所包含的测量与传感功能集。

表99 测量与传感功能集概览

| 功能集标识   | 功能集名称                       | 描述                      |
|---------|-----------------------------|-------------------------|
| 0x0400  | IlluminanceMeasurement      | 用于配置和上报照度测量的属性和命令       |
| 0x0402  | TemperatureMeasurement      | 用于配置和上报温度测量的属性和命令       |
| 0x0403  | PressureMeasurement         | 用于配置和上报压力测量的属性和命令       |
| 0x0404  | FlowMeasurement             | 用于配置和上报流速测量的属性和命令       |
| 0x0405  | RelativeHumidityMeasurement | 支持配置相对湿度测量，并上报空气中相对湿度测量 |
| 0x0406  | OccupancySensing            | 用于配置和上报占用感应状态的属性和命令     |
| various | ResourceMonitoring          | 用于上报各种资源条件的属性和命令        |
| 0x005B  | AirQualityMeasurement       | 用于上报空气质量级别的属性           |
| various | ConcentrationMeasurement    | 浓度测量的属性和别名              |
| 0x005C  | SmokeandCOAlarm             | 提供烟雾和一氧化碳报警接口           |
| 0x0091  | ElectricalEnergyMeasurement | 用于电能测量的属性和命令            |
| 0x0090  | ElectricalPowerMeasurement  | 用于电力测量的属性和命令            |

### 6.3.1.2 测量值

#### 6.3.1.2.1 测量与传感功能集的测量值

介绍MeasuredValue、MinMeasuredValue、MaxMeasuredValue属性的要求。MeasuredValue的精度将在下一节中讨论。

#### 6.3.1.2.2 约束

如果MinMeasuredValue或MaxMeasuredValue属性是强制的，则可以用null值来表示未知的限制。

对于任何具有MeasuredValue、MinMeasuredValue和MaxMeasuredValue属性的测量功能集，应满足以下要求：

- 如果MinMeasuredValue和MaxMeasuredValue都是已知的，则MaxMeasuredValue应大于MinMeasuredValue。
- 如果MaxMeasuredValue已知，则MeasuredValue应小于或等于MaxMeasuredValue。
- 如果MinMeasuredValue已知，则MeasuredValue应大于或等于MinMeasuredValue。

### 6.3.1.3 测量精度

#### 6.3.1.3.1 测量与传感功能集的测量精度

测量功能集可以使用表示简单误差幅度的Tolerance属性来表示其测量的准确性，也可以用表示不同测量值范围的误差幅度或百分比范围的MeasurementAccuracyStruct来表示其测量的准确性。

#### 6.3.1.3.2 Tolerance 属性

对于任何具有MeasuredValue和Tolerance属性的测量功能集，在实现Tolerance时，应满足以下要求：

- Tolerance属性应表示与MeasuredValue属性关联的可能误差的大小（使用相同的单位和分辨率）。真实值应在（MeasuredValue - Tolerance）到（MeasuredValue+Tolerance）的范围内。
- 如果可识别，则真实值永远不应超出可能的物理范围。以下是一些示例：
  - 温度不应低于绝对零度
  - 浓度不应为负值

#### 6.3.1.3.3 MeasurementTypeEnum 类型

此数据类型派生自enum16，详见下表100。

表100 MeasurementTypeEnum 类型表

| 值  | 名称               | 概述                     | 一致性 |
|----|------------------|------------------------|-----|
| 0  | Unspecified      | -                      | M   |
| 1  | Voltage          | 电压 (mV)                | M   |
| 2  | ActiveCurrent    | 有功电流 (mA)              | M   |
| 3  | ReactiveCurrent  | 无功电流 (mA)              | M   |
| 4  | ApparentCurrent  | 视在电流 (mA)              | M   |
| 5  | ActivePower      | 有功功率 (mW)              | M   |
| 6  | ReactivePower    | 无功功率 (mVAR)            | M   |
| 7  | ApparentPower    | 视在功率 (mVA)             | M   |
| 8  | RMSVoltage       | 均方根电压 (mV)             | M   |
| 9  | RMSCurrent       | 均方根电流 (mA)             | M   |
| 10 | RMSPower         | 均方根功率 (mW)             | M   |
| 11 | Frequency        | 交流电频率 (mHz)            | M   |
| 12 | PowerFactor      | 功率因数比 ( $\pm 0.01\%$ ) | M   |
| 13 | NeutralCurrent   | 交流中性线电流 (mA)           | M   |
| 14 | ElectricalEnergy | 电能 (mWh)               | M   |

#### 6.3.1.3.4 MeasurementAccuracyRangeStruct 类型

##### 6.3.1.3.4.1 概述

此结构体表示一系列测量值的测量精度。精度应表示为真实值的最大+/-百分比、真实值的最大+/-固定值，或两者兼而有之，详见下表101。

表101 MeasurementAccuracyRangeStruct 类型结构表

| 标识符 | 名称             | 类型            | 约束条件                   | 质量等级 | 默认 | 通道 | 一致性          |
|-----|----------------|---------------|------------------------|------|----|----|--------------|
| 0   | RangeMin       | int64         | -262to262              | F    | -  | R  | M            |
| 1   | RangeMax       | int64         | -262to262              | F    | -  | R  | M            |
| 2   | PercentMax     | percent100ths | all                    | F    | -  | R  | 0. a+        |
| 3   | PercentMin     | percent100ths | maxPercentTypical      | F    | -  | R  | [PercentMax] |
| 4   | PercentTypical | percent100ths | PercentMintoPercentMax | F    | -  | R  | [PercentMin] |
| 5   | FixedMax       | uint64        | max262-1               | F    | -  | R  | 0. a+        |
| 6   | FixedMin       | uint64        | maxFixedMax            | F    | -  | R  | [FixedMax]   |
| 7   | FixedTypical   | uint64        | FixedMintoFixedMax     | F    | -  | R  | [FixedMin]   |

- 如果同时指定了PercentMax和FixedMax，那么对于在RangeMin和RangeMax之间的给定真实值，
  - 上报的值应小于或等于真实值、FixedMax和真实值的PercentMax百分比之和。
  - 上报的值应大于或等于真实值减去FixedMax和真实值的PercentMax百分比之和。
- 如果仅指定了PercentMax，那么对于RangeMin和RangeMax之间的给定真实值，
  - 上报的值应小于或等于真实值与真实值的PercentMax百分比之和
  - 上报的值应大于或等于真实值减去真实值的PercentMax百分比。
- 如果仅指定了FixedMax，则对于在RangeMin和RangeMax之间的给定真实值，
  - 上报的值应小于或等于真实值与FixedMax之和。
  - 上报的值应大于或等于真实值减去FixedMax。

##### 6.3.1.3.4.2 RangeMin 字段

此字段应指示特定精度级别的最小测量值。

此字段的值应大于或等于MeasurementAccuracyStruct结构体中MinMeasuredValue字段的值。

此字段的值应小于或等于MeasurementAccuracyStruct结构体中MaxMeasuredValue字段的值。

##### 6.3.1.3.4.3 RangeMax 字段

此字段应指示特定精度级别的最大测量值。此段的值应大于RangeMin字段的值。

此字段的值应大于或等于MeasurementAccuracyStruct结构体中的MinMeasuredValue字段的值。  
此字段的值应小于或等于MeasurementAccuracyStruct结构体中的MaxMeasuredValue字段的值。

#### 6.3.1.3.4.4 PercentMax 字段

此字段应指示相关测量的最大+/-百分比精度。

#### 6.3.1.3.4.5 PercentMin 字段

此字段应指示相关测量的最小+/-百分比精度。

#### 6.3.1.3.4.6 PercentTypical 字段

此字段应指示相关测量的典型+/-百分比精度。

#### 6.3.1.3.4.7 FixedMax 字段

此字段应以MeasurementType指定的单位指示相关测量的最大+/-固定精度。

#### 6.3.1.3.4.8 FixedMin 字段

此字段应以MeasurementType指定的单位指示相关测量的最小+/-固定精度。

#### 6.3.1.3.4.9 FixedTypical 字段

此字段应以MeasurementType指定的单位指示相关测量的典型+/-固定精度。

#### 6.3.1.3.5 MeasurementAccuracyStruct 类型

该结构体表示给定测量的精度范围集、测量的最大值和最小值，以及测量是直接测量还是仅根据其  
他信息估算，详见下表102。

表102 MeasurementAccuracyStruct 类型结构表

| 标识符 | 名称               | 类型                                   | 约束条件          | 质量等级 | 默认    | 通道 | 一致性 |
|-----|------------------|--------------------------------------|---------------|------|-------|----|-----|
| 0   | MeasurementType  | MeasurementTypeEnum                  |               | F    |       | R  | M   |
| 1   | Measured         | bool                                 |               | F    | false | R  | M   |
| 2   | MinMeasuredValue | int64                                | -<br>262to262 | F    |       | R  | M   |
| 3   | MaxMeasuredValue | int64                                | -<br>262to262 | F    |       | R  | M   |
| 4   | AccuracyRanges   | List[MeasurementAccuracyRangeStruct] | min1          | F    |       | R  | M   |

其中，

- MeasurementType字段应指示所提供精度的测量类型。
- Measured字段应指示相关测量值是否是直接测量的。如果此字段未设置为true，则表示相关测量是估算的。
- MinMeasuredValueAttribute字段应指示可以测量到的最小值。
- MaxMeasuredValueAttribute字段应指示可以测量到的最大值。
- AccuracyRanges字段应指示测量范围及其相关精度的列表。

此列表中第一个MeasurementAccuracyRangeStruct上的RangeMin字段的值应等于MinMeasuredValue字段的值。

此列表中最后一个MeasurementAccuracyRangeStruct上的RangeMax字段的值应小于或等于MaxMeasuredValue字段的值。

此列表中每个MeasurementAccuracyRangeStruct上的RangeMin字段的值（除第一个外）应比此列表中上一个MeasurementAccuracyRangeStruct上的RangeMax字段的值大1（即，精度范围不应有间隙，量程应依次递增）。

### 6.3.2 照度测量功能集

### 6.3.2.1 概述

照度测量功能集提供了一个用于照度测量功能的接口，包括对照度测量通知进行配置以及提供相关通知。

### 6.3.2.2 分类

该功能集分类详见下表103。

表103 照度测量功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | ILL     |

### 6.3.2.3 Cluster ID

该功能集ID详见下表104。

表104 照度测量功能集 ID 表

| 标识符    | 名称                     |
|--------|------------------------|
| 0x0400 | IlluminanceMeasurement |

### 6.3.2.4 数据类型

LightSensorTypeEnum数据类型是从enum8派生而来，详见下表105。

表105 照度测量功能集数据类型表

| 值       | 名称         | 概述              | 一致性 |
|---------|------------|-----------------|-----|
| 0       | Photodiode | 指示光电二极管传感器类型    | M   |
| 1       | CMOS       | 指示CMOS传感器类型     | M   |
| 64to254 | MS         | 保留给制造商特定的光传感器类型 | 0   |

### 6.3.2.5 属性

#### 6.3.2.5.1 属性列表

该功能集属性详见下表106。

表106 照度测量功能集属性表

| 标识符    | 名称               | 类型                  | 约束条件                                    | 质量等级 | 默认   | 通道 | 一致性 |
|--------|------------------|---------------------|-----------------------------------------|------|------|----|-----|
| 0x0000 | MeasuredValue    | uint16              | 0, MinMeasuredValue to MaxMeasuredValue | PX   | 0    | RV | M   |
| 0x0001 | MinMeasuredValue | uint16              | 1~65533                                 | X    | —    | RV | M   |
| 0x0002 | MaxMeasuredValue | uint16              | min (MinMeasuredValue+1)                | X    | —    | RV | M   |
| 0x0003 | Tolerance        | uint16              | max2048                                 |      | —    | RV | 0   |
| 0x0004 | LightSensorType  | LightSensorTypeEnum | all                                     | X    | null | RV | 0   |

#### 6.3.2.5.2 MeasuredValue 属性

该属性应以Lux（符号lx）指示照度（illuminance），如下公式（1）所示：

$$MeasuredValue = 10,000 \times \log_{10} (illuminance) + 1 \dots\dots\dots (1)$$

式中：1lx≤illuminance≤3.576Mlx，对应于1到0xFFFF范围内的MeasuredValue。

MeasuredValue属性可以采用以下值：

0指示照度值太低而无法测量，在正常情况下，MinMeasuredValue <= MeasuredValue <= MaxMeasuredValue，null指示照度测量无效。

MeasuredValue属性在进行新的测量时会持续更新。

6.3.2.5.3 MinMeasuredValue 属性

该属性应指示可测量的MeasuredValue的最小值。取null值指示没有定义该属性。有关详细信息，请参见测量值。

6.3.2.5.4 MaxMeasuredValue 属性

该属性应指示可测量的MeasuredValue的最大值。取null值指示没有定义该属性。有关详细信息，请参见测量值。

6.3.2.5.5 Tolerance 属性

见测量值。

6.3.2.5.6 LightSensorType 属性

该属性应指示光传感器的电子类型。此属性应设置为LightSensorTypeEnum中列出的非保留值之一，如果传感器类型未知，则设置为null。

6.3.3 温度测量功能集

6.3.3.1 概述

该功能集提供了一个用于温度测量功能的接口，包括对温度测量通知进行配置以及提供此类通知。

6.3.3.2 分类

该功能集分类详见下117。

表107 温度测量功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | TMP     |

6.3.3.3 Cluster ID

该功能集ID详见下表108。

表108 温度测量功能集 ID 表

| 标识符    | 名称                     |
|--------|------------------------|
| 0x0402 | TemperatureMeasurement |

6.3.3.4 属性

该功能集属性详见下表109。

表109 温度测量功能集属性表

| 标识符    | 名称               | 类型          | 约束条件                                 | 质量等级 | 默认 | 通道 | 一致性 |
|--------|------------------|-------------|--------------------------------------|------|----|----|-----|
| 0x0000 | MeasuredValue    | temperature | MinMeasuredValue to MaxMeasuredValue | XP   | -  | RV | M   |
| 0x0001 | MinMeasuredValue | temperature | -27315~32766                         | X    | -- | RV | M   |
| 0x0002 | MaxMeasuredValue | temperature | min (MinMeasuredValue+1)             | X    |    | RV | M   |
| 0x0003 | Tolerance        | uint16      | max2048                              | -    | 0  | RV | O   |

其中，

- MeasuredValue属性应指示测量的温度。null值指示温度未知。

- MinMeasuredValue属性应指示能够测量的MeasuredValue的最小值。有关详细信息，请参见测量值。null值指示该值不可用。
- MaxMeasuredValue属性指示可测量的MeasuredValue的最大值。有关详细信息，请参见测量值。null值指示该值不可用。
- Tolerance属性见测量值。

### 6.3.4 含水量测量功能集

#### 6.3.4.1 概述

这是一个基础功能集。服务端功能集提供了一个用于含水量测量功能的接口。该测量结果可进行报告，并且可以针对报告进行相应配置。目前，含水量测量包含但不限于相对湿度这一项内容。

#### 6.3.4.2 分类

该功能集分类详见下表110。

表110 含水量测量功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | RH      |

#### 6.3.4.3 Cluster ID

该功能集ID详见下表111。

表111 含水量测量功能集 ID 表

| 标识符    | 名称                          |
|--------|-----------------------------|
| 0x0405 | RelativeHumidityMeasurement |

#### 6.3.4.4 属性

该功能集属性详见下表112。

表112 含水量测量功能集属性表

| 标识符    | 名称               | 类型     | 约束条件                                 | 质量等级 | 默认 | 通道 | 一致性 |
|--------|------------------|--------|--------------------------------------|------|----|----|-----|
| 0x0000 | MeasuredValue    | uint16 | MinMeasuredValue to MaxMeasuredValue | XP   | —  | RV | M   |
| 0x0001 | MinMeasuredValue | uint16 | max9999                              | X    | —  | RV | M   |
| 0x0002 | MaxMeasuredValue | uint16 | (MinMeasuredValue+1)~10000           | X    | —  | RV | M   |
| 0x0003 | Tolerance        | uint16 | max2048                              | —    | —  | RV | O   |

其中，

- MeasuredValue 属性表示以%为单位的含水量（watercontent），如下所示：  
MeasuredValue=100xwatercontent当含水量在0%到100%之间时，相应的MeasuredValue在0到10000 的范围内，这种格式允许的最大分辨率是0.01%。MinMeasuredValue 和 MaxMeasuredValue定义了传感器的测量范围。null值表示测量结果未知，否则其范围应如测量值所定义。MeasuredValue会随着新的测量不断更新。
- MinMeasuredValue属性表示可测量的MeasuredValue的最小值。null值意味着该属性未定义。更多详细信息，请参阅测量值。
- MaxMeasuredValue属性表示可测量的MeasuredValue的最大值。null值意味着该属性未定义。更多详细信息，请参阅测量值。
- Tolerance属性详见测量值。

### 6.3.5 占位感应功能集

#### 6.3.5.1 概述

服务端功能集提供了一个基于一种或多种感应模式的占位感应功能接口，包括对占位状态通知进行配置以及提供相关通知。

### 6.3.5.2 分类

该功能集分类详见下表113。

表113 占位感应功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | OCC     |

### 6.3.5.3 Cluster ID

该功能集ID详见下表114。

表114 占位感应功能集 ID 表

| 标识符    | 名称               |
|--------|------------------|
| 0x0406 | OccupancySensing |

### 6.3.5.4 特性

这个功能集应支持如下定义的FeatureMapbitmap属性，详见表115。

表115 占位感应功能集位图属性表

| Bit | 编码    | 特征              | 一致性  | 概述                                   |
|-----|-------|-----------------|------|--------------------------------------|
| 0   | OTHER | Other           | 0.a+ | 支持使用其他位中未列出的方式进行感应                   |
| 1   | PIR   | PassiveInfrared | 0.a+ | 支持使用PIR（被动红外）方式进行感应                  |
| 2   | US    | Ultrasonic      | 0.a+ | 支持使用超声波方式进行感应                        |
| 3   | PHY   | PhysicalContact | 0.a+ | 支持使用物理接触方式进行感应                       |
| 4   | AIR   | ActiveInfrared  | 0.a+ | 支持使用主动红外测量方式进行感应（如，飞行时间或半透反射/反射红外传感） |
| 5   | RAD   | Radar           | 0.a+ | 支持使用雷达波（微波）方式进行感应                    |
| 6   | RFS   | RFSensing       | 0.a+ | 支持基于射频信号分析的感应                        |
| 7   | VIS   | Vision          | 0.a+ | 支持基于图像分析的感应                          |

### 6.3.5.5 数据类型

#### 6.3.5.5.1 Occupancy Bitmap 类型

该数据类型派生自map8，详见下表116。

表116 Occupancy Bitmap 类型表

| Bit | 名称       | 概述       | 一致性 |
|-----|----------|----------|-----|
| 0   | Occupied | 表示感应占用状态 | M   |

其中，如果设置了Occupied位，则它应指示占用状态，如果未设置该位，则应指示未占用状态。

#### 6.3.5.5.2 OccupancySensorTypeBitmap 类型

该数据类型派生自map8，详见下表117。

表117 OccupancySensorTypeBitmap 类型表

| Bit | 名称              | 概述         | 一致性 |
|-----|-----------------|------------|-----|
| 0   | PIR             | 表示被动红外传感器  | M   |
| 1   | Ultrasonic      | 表示超声波传感器   | M   |
| 2   | PhysicalContact | 表示物理接触式传感器 | M   |

注：此枚举在ClusterRevision4中所定义，其定义不应扩展；特征标志为后续的功能集修订版提供传感器模式（或模式组合）。

### 6.3.5.5.3 OccupancySensorTypeEnum 类型

该数据类型派生自enum8，详见下表118。

表118 OccupancySensorTypeEnum 类型表

| 值 | 名称               | 概述            | 一致性 |
|---|------------------|---------------|-----|
| 0 | PIR              | 表示被动红外传感器     | M   |
| 1 | Ultrasonic       | 表示超声波传感器      | M   |
| 2 | PIRAndUltrasonic | 表示被动红外和超声波传感器 | M   |
| 3 | PhysicalContact  | 表示物理接触式传感器    | M   |

注：此枚举在ClusterRevision4中所定义，其定义不应扩展；特征标志为后续的功能集修订版提供传感器模式（或模式组合）。

### 6.3.5.5.4 HoldTimeLimitsStruct 类型

此结构提供有关服务端支持的HoldTime属性值的信息，详见下表119。

表119 HoldTimeLimitsStruct 类型表

| 标识符 | 名称              | 类型     | 约束条件                      | 质量等级 | 默认 | 通道 | 一致性 |
|-----|-----------------|--------|---------------------------|------|----|----|-----|
| 0   | HoldTimeMin     | uint16 | min1                      | —    | —  | —  | M   |
| 1   | HoldTimeMax     | uint16 | min (HoldTimeMin) , min10 | —    | —  | —  | M   |
| 2   | HoldTimeDefault | uint16 | HoldTimeMintoHoldTimeMax  | —    | —  | —  | M   |

其中，

- HoldTimeMin字段应指示服务端支持的HoldTime属性的最小值，以秒为单位。
- HoldTimeMax字段应指示服务端支持的HoldTime属性的最大值，以秒为单位。
- HoldTimeDefault字段应指示服务端的HoldTime属性（制造商定义的）的默认值（以秒为单位）。这是一个客户端想要将设置重置为有效默认值时建议使用的值。

### 6.3.5.6 属性

#### 6.3.5.6.1 属性列表

该功能集属性结构详见下表120。

表120 占位感应功能集属性结构表

| 标识符    | 名称                           | 类型                        | 约束条件 | 质量等级 | 默认 | 通道   | 一致性                                                                                                              |
|--------|------------------------------|---------------------------|------|------|----|------|------------------------------------------------------------------------------------------------------------------|
| 0x0000 | Occupancy                    | OccupancyBitmap           | 0~1  | P    | —  | RV   | M                                                                                                                |
| 0x0001 | OccupancySensorType          | OccupancySensorTypeEnum   | desc | F    | —  | RV   | M, D                                                                                                             |
| 0x0002 | OccupancySensorTypeBitmap    | OccupancySensorTypeBitmap | 0~7  | F    | —  | RV   | M, D                                                                                                             |
| 0x0003 | HoldTime                     | uint16                    | desc | N    | —  | RWVM | 0                                                                                                                |
| 0x0004 | HoldTimeLimits               | HoldTimeLimitsStruct      |      | F    | —  | RV   | HoldTime                                                                                                         |
| 0x0010 | PIROccupiedToUnoccupiedDelay | uint16                    | all  | N    | 0  | RWVM | [HoldTime& (PIR  (!PIR&!US&!PHY) ) ], D                                                                          |
| 0x0011 | PIRUnoccupiedToOccupiedDelay | uint16                    | all  | N    | 0  | RWVM | (HoldTime& (PIR  (!PIR&!US&!PHY) ) ) & PIRUnoccupiedToOccupiedThreshold, [HoldTime& (PIR  (!PIR&!US&!PHY) ) ], D |

表120 占位感应功能集属性结构表（续）

| 标识符    | 名称                                           | 类型     | 约束条件  | 质量等级 | 默认 | 通道   | 一致性                                                                                                |
|--------|----------------------------------------------|--------|-------|------|----|------|----------------------------------------------------------------------------------------------------|
| 0x0012 | PIRUnoccupiedToOccupiedThreshold             | uint8  | 1~254 | N    | 1  | RWVM | (HoldTime&(PIR (!PIR&!US&!PHY)))&PIRUnoccupiedToOccupiedDelay, [HoldTime&(PIR (!PIR&!US&!PHY))], D |
| 0x0020 | UltrasonicOccupiedToUnoccupiedDelay          | uint16 | all   | N    | 0  | RWVN | [HoldTime&US], D                                                                                   |
| 0x0021 | UltrasonicUnoccupiedToOccupiedDelay          | uint16 | all   | N    | 0  | RWVN | HoldTime&US&UltrasonicUnoccupiedToOccupiedThreshold, [HoldTime&US], D                              |
| 0x0022 | UltrasonicUnoccupiedToOccupiedThreshold      | uint8  | 1~254 | N    | 1  | RWVN | HoldTime&US&UltrasonicUnoccupiedToOccupiedDelay, [HoldTime&US], D                                  |
| 0x0030 | PhysicalContactOccupiedToUnoccupiedDelay     | uint16 | all   | N    | 0  | RWVN | [HoldTime&PHY], D                                                                                  |
| 0x0031 | PhysicalContactUnoccupiedToOccupiedDelay     | uint16 | all   | N    | 0  | RWVN | HoldTime&PHY&PhysicalContactUnoccupiedToOccupiedThreshold, [HoldTime&PHY], D                       |
| 0x0032 | PhysicalContactUnoccupiedToOccupiedThreshold | uint8  | 1~254 | N    | 1  | RWVN | HoldTime&PHY&PhysicalContactUnoccupiedToOccupiedDelay, [HoldTime&PHY], D                           |

6.3.5.6.2 Occupancy 属性

该属性应指示占用的感应（处理）状态。出于兼容性考虑，使用位图表示状态，其中第0位表示状态：值为1表示占用，值为0表示未占用，其余位设置为0；这可以视为boolean值。

6.3.5.6.3 OccupancySensorType 和 OccupancySensorTypeBitmap 属性

这些属性应指示传感器的类型。在ClusterRevision<=4的情况下，这些属性用于通过位图指示占用传感器的类型。而在ClusterRevision>=5的情况下，服务端应使用featureflags指示传感器的类型，并且服务端应根据下表映射提供这些属性，以便与实现了ClusterRevision<=4的客户端保持向后兼容，详见下表121。

表121 属性映射表

| FeatureFlag值 |    |     | OccupancySensorTypeBitmap值     | OccupancySensorType值 |
|--------------|----|-----|--------------------------------|----------------------|
| PIR          | US | PHY | —                              |                      |
| 0            | 0  | 0   | PIR*                           | PIR*                 |
| 1            | 0  | 0   | PIR                            | PIR                  |
| 0            | 1  | 0   | Ultrasonic                     | Ultrasonic           |
| 1            | 1  | 0   | PIR+Ultrasonic                 | PIRAndUltrasonic     |
| 0            | 0  | 1   | PhysicalContact                | PhysicalContact      |
| 1            | 0  | 1   | PhysicalContact+PIR            | PIR**                |
| 0            | 1  | 1   | PhysicalContact+Ultrasonic     | Ultrasonic**         |
| 1            | 1  | 1   | PhysicalContact+PIR+Ultrasonic | PIRAndUltrasonic**   |

如果功能标志PIR、US和PHY都设置为0，则这两个属性将设置为虚拟的PIR传感器，因为这能为实现了ClusterRevision<=4但无法识别功能标志和新增的模式模式的客户端提供最佳兼容性。在这种情况下，定时参数（当支持HoldTime时）使用PIROccupiedToUnoccupiedDelay、PIRUnoccupiedToOccupiedDelay和PIRUnoccupiedToOccupiedThreshold属性表示。

这些传感器类型的组合没有在ClusterRevision $\leq 4$ 中被OccupancySensorType定义，因此它们被映射到该版本中定义的组合。

#### 6.3.5.6.4 HoldTime 属性

此属性应指示传感器在最后一次检测到感应区域的占用状态后，转换为未占用状态之前的延迟时间（以秒为单位）。这相当于旧版的OccupiedToUnoccupiedDelay属性。

HoldTime 的值应在 HoldTimeLimits 属性中提供的限制范围内，即  $\text{HoldTimeMin} \leq \text{HoldTime} \leq \text{HoldTimeMax}$ 。建议避免使用较低的HoldTime，因为它们可能会导致大量的上报消息。不应使HoldTime的值为0。

下图18通过一个示例来说明如何将此属性用于PIR传感器。它使用阈值检测来生成“内部检测”信号，该信号需要后处理才能用于传输（流量整形）。当内部检测信号变为高电平时，Occupancy属性中的位将设置为1，并将在内部检测信号变低后的HoldTime时间内保持为1。

图的上半部分显示了单次触发的情况：Occupancy属性中的位将在PIR信号超过阈值并持续HoldTime的时间内保持为1。

图的下半部分显示了多次触发的情况：第二次触发发生在第一次触发的HoldTime到期之前；这将导致Occupancy属性中的位在一个连续期间内保持为1。Occupancy属性中的位将在PIR信号首次超过阈值的周期开始时设置为1，直到PIR信号最后一次超过阈值后HoldTime时间结束。

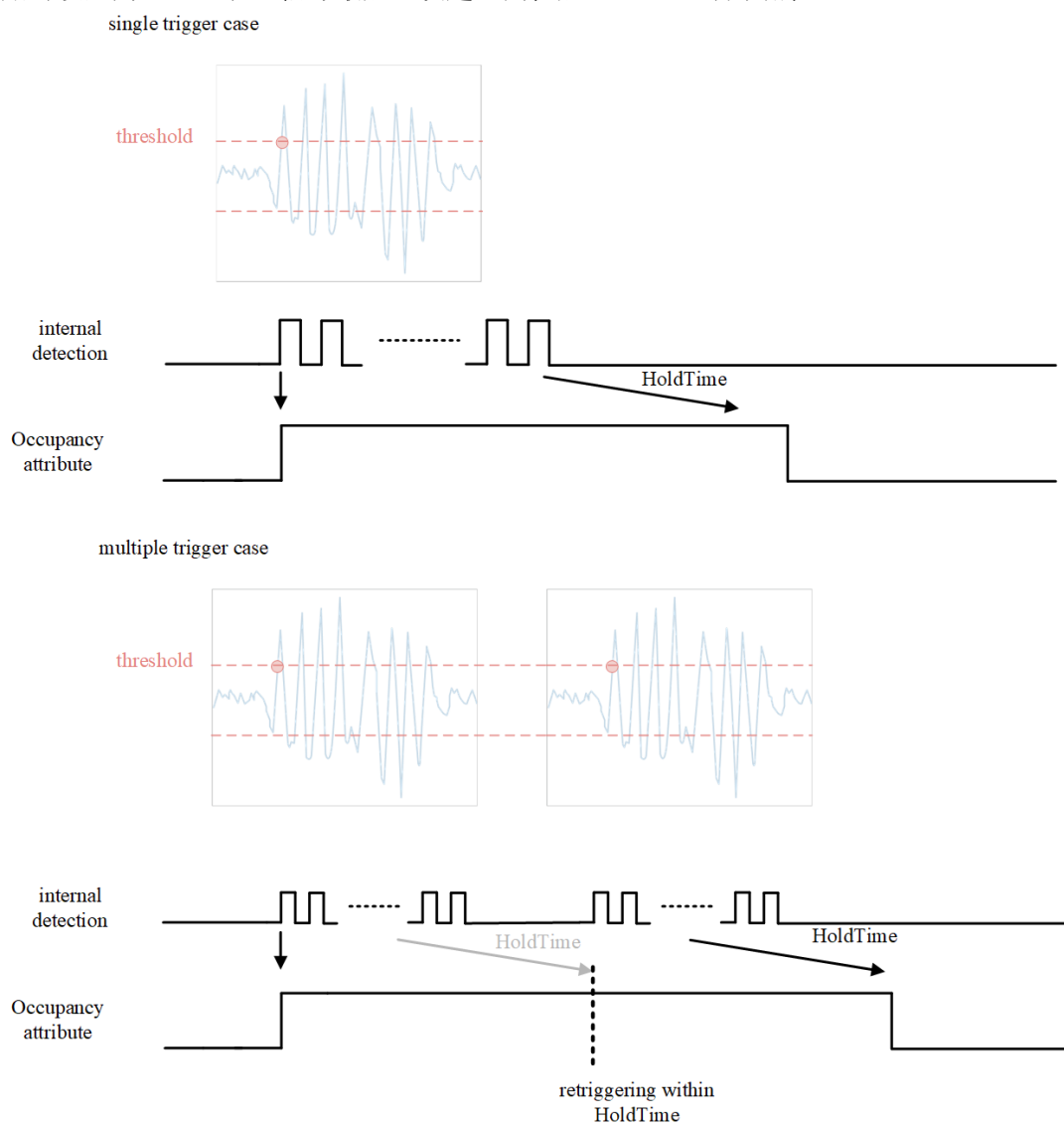


图18 使用 HoldTime 处理指向 Occupancy 属性的 PIR 信号

### 6.3.5.6.5 HoldTimeLimits 属性

此属性应指示服务端对HoldTime属性的限制和默认值。

### 6.3.5.6.6 支持传统实现的属性

以下三个属性集，每个包含三个属性，在ClusterRevision<=4中用于指示调整传感器上报的时间和条件——分别用于该修订版中定义的三种传感器模式。在ClusterRevision>=5中，这些被HoldTime属性替代（该属性对所有模式共享）。ClusterRevision>=5的服务端可以（在上面属性表中列出的符合性条件下）为与实现ClusterRevision<=4的客户端保持向后兼容而提供这些属性。在这样做时，提供的这些旧的\*OccupiedToUnoccupied属性的值在更新时应遵循HoldTime属性的值，反之亦然，确保任何现有的\*OccupiedToUnoccupiedDelay属性与HoldTime保持相等。

一致性规则对这些属性的部分（但不是全部）影响包括：

- 如果不支持HoldTime属性，则不应支持这9个旧属性
- 如果支持PIR模式，但不支持PHY和US模式：
  - 与PIR相关的3个旧属性可以被支持（如果其他规则允许）。
  - 与Ultrasonic和PhysicalContact相关的6个旧属性不应被支持。
- 如果支持US模式，但不支持PIR和PHY模式：
  - 与Ultrasonic相关的3个旧属性可以被支持（如果其他规则允许）。
  - 与PIR和PhysicalContact相关的6个旧属性不应被支持。
- 如果支持PHY模式，但不支持PIR和US模式：
  - 与PhysicalContact相关的3个旧属性可以被支持（如果其他规则允许）。
  - 与PIR和Ultrasonic相关的6个旧属性不应被支持。
- 与UnoccupiedToOccupiedDelay和UnoccupiedToOccupiedThreshold相关的旧属性对于某一模式，应该要么都支持，要么都不支持。
- 如果不支持PIR、US和PHY中的任何模式（此情况作为PIR传感器进行兼容处理）：
  - 与PIR相关的3个旧属性可以被支持（如果其他规则允许）。
  - 与Ultrasonic和PhysicalContact相关的6个旧属性不应被支持。

对于支持多个PIR、PHY和US模式的情况，并且为多个模式提供了时间参数，如何组合这些定时参数是制造商定义的。

### 6.3.5.6.7 PIROccupiedToUnoccupiedDelay 属性

该属性应指示在感应区域内最后一次检测到占用后，PIR传感器转为未占用状态之前的延迟时间（以秒为单位）。

### 6.3.5.6.8 PIRUnoccupiedToOccupiedDelay 属性

该属性应指示在感应区域内第一次检测到占用后，PIR传感器转为占用状态之前的延迟时间（以秒为单位）。

### 6.3.5.6.9 PIRUnoccupiedToOccupiedThreshold 属性

该属性应指示在PIR传感器转换为占用状态之前，在PIRUnoccupiedToOccupiedDelay期间必须发生的占用检测事件的次数。

### 6.3.5.6.10 UltrasonicOccupiedToUnoccupiedDelay 属性

该属性应指示在感应区域内最后一次检测到占用后，超声波传感器转为未占用状态之前的延迟时间（以秒为单位）。

### 6.3.5.6.11 UltrasonicUnoccupiedToOccupiedDelay 属性

该属性应指示在感应区域内第一次检测到占用后，超声波传感器转为占用状态之前的延迟时间（以秒为单位）。

#### 6.3.5.6.12 UltrasonicUnoccupiedToOccupiedThreshold 属性

该属性应指示在超声波传感器转换为占用状态之前，在UltrasonicUnoccupiedToOccupiedDelay期间必须发生的占用检测事件的次数。

#### 6.3.5.6.13 PhysicalContactOccupiedToUnoccupiedDelay 属性

该属性应指示在感应区域内最后一次检测到占用后，物理接触占位传感器转为未占用状态之前的延迟时间（以秒为单位）。

#### 6.3.5.6.14 PhysicalContactUnoccupiedToOccupiedDelay 属性

该属性应指示在感应区域内第一次检测到占用后，物理接触占位传感器转为占用状态之前的延迟时间（以秒为单位）。

#### 6.3.5.6.15 PhysicalContactUnoccupiedToOccupiedThreshold 属性

该属性应指示在物理接触占位传感器转换为占用状态之前，在PhysicalContactUnoccupiedToOccupiedDelay期间必须发生的占用检测事件的次数。

#### 6.3.5.7 事件

该功能集事件详见下表122。

表122 占位感应功能集事件表

| 标识符  | 名称               | 重要等级 | 质量等级 | 通道 | 一致性 |
|------|------------------|------|------|----|-----|
| 0x00 | OccupancyChanged | INFO |      | V  | 0   |

如果支持OccupancyChanged事件，则应在Occupancy属性变化时生成该事件，详见下表123。

表123 OccupancyChanged 事件表

| 标识符 | 名称        | 类型              | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------|-----------------|------|------|----|-----|
| 0   | Occupancy | OccupancyBitmap | —    | —    | —  | M   |

其中，Occupancy字段应指示Occupancy属性的新值。

#### 6.3.6 空气质量功能集

##### 6.3.6.1 概述

该功能集为空气质量分类提供了一个使用易读标签的不同等级的接口。

##### 6.3.6.2 分类

该功能集分类详见下表124。

表124 空气质量功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | AIRQUAL |

##### 6.3.6.3 Cluster ID

该功能集ID详见下表125。

表125 空气质量功能集 ID 表

| 标识符    | 名称         |
|--------|------------|
| 0x005B | AirQuality |

##### 6.3.6.4 特性

该功能集应支持如下定义的FeatureMapbitmap属性，详见下表126。

表126 空气质量功能集位图属性表

| Bit | 编码    | 特征            | 一致性 | 概述             |
|-----|-------|---------------|-----|----------------|
| 0   | FAIR  | Fair          | 0   | 功能集支持一般空气质量级别  |
| 1   | MOD   | Moderate      | 0   | 功能集支持中度空气质量级别  |
| 2   | VPOOR | VeryPoor      | 0   | 功能集支持空气质量非常差级别 |
| 3   | XPOOR | ExtremelyPoor | 0   | 功能集支持空气质量极差级别  |

6.3.6.5 数据类型

AirQualityEnum数据类型源自enum8。AirQualityEnum提供了分析空气质量的表示方式。设备制造商可自行决定测量值与相应枚举值之间的映射关系，详见下表127。

表127 空气质量功能集测量值映射表

| 值 | 名称            | 概述      | 一致性   |
|---|---------------|---------|-------|
| 0 | Unknown       | 空气质量未知  | M     |
| 1 | Good          | 空气质量良好  | M     |
| 2 | Fair          | 空气质量一般  | FAIR  |
| 3 | Moderate      | 空气质量中等  | MOD   |
| 4 | Poor          | 空气质量差   | M     |
| 5 | VeryPoor      | 空气质量非常差 | VPOOR |
| 6 | ExtremelyPoor | 空气质量极差  | XPOOR |

6.3.6.6 属性

空气质量信息属性集包含下表汇总的属性，详见下表128。

表128 空气质量功能集属性表

| 标识符    | 名称         | 类型             | 约束条件 | 质量等级 | 默认 | 通道 | 一致性 |
|--------|------------|----------------|------|------|----|----|-----|
| 0x0000 | AirQuality | AirQualityEnum | desc | -    | 0  | RV | M   |

其中，AirQuality属性应指示标明当前测量的空气质量的AirQualityEnum中的一个值。

6.3.7 浓度测量功能集

6.3.7.1 概述

该服务端功能集为浓度测量功能提供了一个接口。应通过特定物质的别名使用该功能集（见Cluster ID）。

6.3.7.2 分类

该功能集分类详见下表129。

表129 浓度测量功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | CONC    |

6.3.7.3 Cluster ID

下表130是符合本文件的Cluster ID列表。

表130 浓度测量功能集 ID 表

| 标识符    | Name          | PICSCode |
|--------|---------------|----------|
| 0x040C | 一氧化碳浓度测量      | CMOCONC  |
| 0x040D | 二氧化碳浓度测量      | CDOCONC  |
| 0x0413 | 二氧化氮浓度测量      | NDOCONC  |
| 0x0415 | 臭氧浓度测量        | OZCONC   |
| 0x042A | PM2.5浓度测量     | PMICONC  |
| 0x042B | 甲醛浓度测量        | FLDCONC  |
| 0x042C | PM1浓度测量       | PMHCONC  |
| 0x042D | PM10浓度测量      | PMKCONC  |
| 0x042E | 总挥发性有机化合物浓度测量 | TVOCCONC |
| 0x042F | 氢浓度测量         | RNCONC   |

#### 6.3.7.4 特性

该功能集应支持如下定义的FeatureMapbitmap属性，详见下表131。

表131 浓度测量功能集位图属性表

| Bit | 编码  | 特征                 | 一致性   | 概述                                     |
|-----|-----|--------------------|-------|----------------------------------------|
| 0   | MEA | NumericMeasurement | 0. a+ | 功能集支持对物质进行数字测量                         |
| 1   | LEV | LevelIndication    | 0. a+ | 功能集支持使用ConcentrationLevel枚举对物质进行基本级别指示 |
| 2   | MED | MediumLevel        | [LEV] | 功能集支持Medium Concentration Level        |
| 3   | CRI | CriticalLevel      | [LEV] | 功能集支持CriticalConcentrationLevel        |
| 4   | PEA | PeakMeasurement    | [MEA] | 功能集支持对物质进行峰值数值测量                       |
| 5   | AVG | AverageMeasurement | [MEA] | Cluster支持对物质进行平均数值测量                   |

#### 6.3.7.5 数据类型

##### 6.3.7.5.1 MeasurementUnitEnum 类型

该数据类型源自enum8，详见下表132。

表132 MeasurementUnitEnum 类型表

| 值 | 名称   | 概述         | 一致性 |
|---|------|------------|-----|
| 0 | PPM  | 百万分率（106）  | MEA |
| 1 | PPB  | 十亿分率（109）  | MEA |
| 2 | PPT  | 万亿分率（1012） | MEA |
| 3 | MGM3 | 毫克/m3      | MEA |
| 4 | UGM3 | 微克/m3      | MEA |
| 5 | NGM3 | 纳克/m3      | MEA |
| 6 | PM3  | 颗粒数/m3     | MEA |
| 7 | BQM3 | 贝克勒尔/m3    | MEA |

其中提到的“十亿”指109，“万亿”指1012（短标尺）。

##### 6.3.7.5.2 MeasurementMediumEnum 类型

该数据类型源自enum8，详见下表133。

表133 MeasurementMediumEnum 类型表

| 值 | 名称    | 概述       | 一致性 |
|---|-------|----------|-----|
| 0 | Air   | 测量在空气中进行 | M   |
| 1 | Water | 测量在水中进行  | M   |
| 2 | Soil  | 测量在土壤中进行 | M   |

6.3.7.5.3 LevelValueEnum 类型

该数据类型源自enum8，详见下表134。

表134 LevelValueEnum 类型表

| 值 | 名称       | 概述         | 一致性 |
|---|----------|------------|-----|
| 0 | Unknown  | 级别未知       | M   |
| 1 | Low      | 该级别被视为“低”  | M   |
| 2 | Medium   | 该级别被视为“中等” | MED |
| 3 | High     | 该级别被视为“高”  | M   |
| 4 | Critical | 该级别被视为“临界” | CRI |

6.3.7.6 属性

该功能集属性详见下表135。

表135 浓度测量功能集属性表

| 标识符    | 名称                         | 类型                    | 约束条件                                 | 质量等级 | 默认   | 通道 | 一致性   |
|--------|----------------------------|-----------------------|--------------------------------------|------|------|----|-------|
| 0x0000 | MeasuredValue              | single                | MinMeasuredValue to MaxMeasuredValue | XP   | null | RV | MEA   |
| 0x0001 | MinMeasuredValue           | single                | all                                  | X    | null | RV | MEA   |
| 0x0002 | MaxMeasuredValue           | single                | minMinMeasuredValue                  | X    | null | RV | MEA   |
| 0x0003 | PeakMeasuredValue          | single                | MinMeasuredValue to MaxMeasuredValue | XP   | null | RV | PEA   |
| 0x0004 | PeakMeasuredValueWindow    | elapsed-s             | Max604800                            | P    | 1    | RV | PEA   |
| 0x0005 | AverageMeasuredValue       | single                | MinMeasuredValue to MaxMeasuredValue | XP   | null | RV | AVG   |
| 0x0006 | AverageMeasuredValueWindow | elapsed-s             | Max604800                            | P    | 1    | RV | AVG   |
| 0x0007 | Uncertainty                | single                | MS                                   | --   | MS   | RV | [MEA] |
| 0x0008 | MeasurementUnit            | MeasurementUnitEnum   | —                                    | F    | MS   | RV | MEA   |
| 0x0009 | MeasurementMedium          | MeasurementMediumEnum | —                                    | F    | MS   | RV | M     |
| 0x000A | LevelValue                 | LevelValueEnum        | —                                    | —    | 0    | RV | LEV   |

其中，

- MeasuredValue属性应将最近的测量值表示为单精度浮点数。MeasuredValue的单位由MeasurementUnit表示。null表示测量值未知或超出有效范围。MinMeasuredValue（最小测量值）和MaxMeasuredValue（最大测量值）定义了MeasuredValue的有效范围。
- MinMeasuredValue属性应表示可测量的MeasuredValue的最小值。MinMeasuredValue为null表示未定义MinMeasuredValue。
- MaxMeasuredValue属性应表示可测量的MeasuredValue的最大值。MaxMeasuredValue为null表示未定义MaxMeasuredValue。
- PeakMeasuredValue属性应表示在PeakMeasuredValueWindow（峰值测量值窗口）期间测量到的最大测量值。如果提供此属性，则还应提供PeakMeasuredValueWindow属性。
- PeakMeasuredValueWindow属性应表示用于确定PeakMeasuredValue的时间窗口。单位为秒。
- AverageMeasuredValue属性应表示在AverageMeasuredValueWindow（平均测量值窗口）期间测量的MeasuredValue的平均值。如果提供此属性，则还应提供AverageMeasuredValueWindow属性。
- AverageMeasuredValueWindow属性应表示用于确定AverageMeasuredValue的时间窗口。单位为秒。
- Uncertainty属性应表示测量值和峰值测量值之间可找到的误差或偏差范围。这被视为+/-值，且应考虑在测量单位（MeasurementUnit）内。
- MeasurementUnit属性应表示MeasuredValue（测量值）的单位。请参见“6.4.8.6.1 MeasurementUnitEnum类型”。

- MeasurementMedium属性应表示测量MeasuredValue的中间值。请参阅“6.4.8.6.2 MeasurementMediumEnum类型”。
- LevelValue属性应表示检测到的物质的含量水平。请参见“6.4.8.6.3 LevelValueEnum类型”。

6.3.8 烟雾一氧化碳警报功能集

6.3.8.1 概述

该功能集为观察和管理烟雾报警器和一氧化碳报警器的状态提供了一个接口。

6.3.8.2 分类

该功能集分类详见下表136。

表136 烟雾一氧化碳警报功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | SMOKECO |

6.3.8.3 Cluster ID

该功能集ID详见下表137。

表137 烟雾一氧化碳警报功能集 ID 表

| 标识符    | 名称           |
|--------|--------------|
| 0x005C | SmokeCOAlarm |

6.3.8.4 特性

该功能集应支持如下定义的FeatureMapbitmap属性，详见下表138。

表138 烟雾一氧化碳警报功能集位图属性表

| Bit | 编码    | 特征         | 一致性   | 概述       |
|-----|-------|------------|-------|----------|
| 0   | SMOKE | SmokeAlarm | 0, a+ | 支持烟雾报警   |
| 1   | CO    | COAlarm    | 0, a+ | 支持一氧化碳报警 |

6.3.8.5 数据类型

6.3.8.5.1 AlarmStateEnum 类型

该数据类型源自enum8，详见下表139。

表139 AlarmStateEnum 类型表

| 值 | 名称       | 概述         | 一致性 |
|---|----------|------------|-----|
| 0 | Normal   | 标称状态，设备不报警 | M   |
| 1 | Warning  | 警告状态       | O   |
| 2 | Critical | 临界状态       | M   |

其中，

- Normal值应表示该报警器未发出报警。
- Warning值应表示该报警器处于警告状态。处于此状态的报警器应可通过物理交互方式静音。
- Critical值应表示该报警器处于临界状态。处于该状态的报警器不得通过物理交互方式静音。此值应表示此警报处于严重状态。处于此状态的警报不得通过物理交互静音。

6.3.8.5.2 SensitivityEnum 类型

该数据类型源自enum8，详见下表140。

表140 SensitivityEnum 类型表

| 值 | 名称       | 概述    | 一致性 |
|---|----------|-------|-----|
| 0 | High     | 高灵敏度  | O   |
| 1 | Standard | 标准灵敏度 | M   |
| 2 | Low      | 低灵敏度  | O   |

6.3.8.5.3 ExpressedStateEnum 类型

该数据类型源自enum8，详见下表141。

表141 ExpressedStateEnum 类型表

| 值 | 名称                | 概述         | 一致性   |
|---|-------------------|------------|-------|
| 0 | Normal            | 标称状态，设备不报警 | M     |
| 1 | SmokeAlarm        | 烟雾报警器状态    | SMOKE |
| 2 | COAlarm           | 一氧化碳警报状态   | CO    |
| 3 | BatteryAlert      | 电池警报状态     | M     |
| 4 | Testing           | 测试进行中      | M     |
| 5 | HardwareFault     | 硬件故障警报状态   | M     |
| 6 | EndOfService      | 服务终止警报状态   | M     |
| 7 | InterconnectSmoke | 互联烟雾报警器状态  | O     |
| 8 | InterconnectCO    | 互联一氧化碳警报状态 | O     |

其中，

- Normal值应表示该报警器未发出报警。
- SmokeAlarm值应表示该报警器当前显示烟雾报警的视觉指示。除非支持DeviceMuted属性并将其设置为静音，否则该值应表示报警器当前正在发出烟雾警报的声音指示。
- COAlarm值应表示该报警器当前显示一氧化碳报警的视觉指示。除非支持DeviceMuted属性并将其设置为静音，否则该值应表示报警器当前正在发出一氧化碳报警的声音指示。
- BatteryAlert值应表示该警报当前显示电池电量严重不足的视觉指示。除非支持DeviceMuted属性并将其设置为静音，否则该值应表示报警器当前正在发出“电池电量严重不足”的声音指示。
- Testing值应表示该报警器当前显示自检的视觉和听觉指示。
- HardwareFault值应表示该报警器当前显示硬件故障的视觉指示。除非支持DeviceMuted属性并将其设置为静音，否则该值应表示报警器当前正在发出硬件故障的声音指示。
- EndOfService值应表示该报警器当前显示服务终止的视觉指示。除非支持DeviceMuted属性并将其设置为静音InterconnectSmoke值。该值应表示该报警器当前显示由互联引起的烟雾警报的视觉指示。除非支持DeviceMuted属性并将其设置为静音，否则该值应表示报警器当前表示由互连引起的烟雾报警的声音指示。
- InterconnectCO值应表示该报警器当前显示由互联引起的一氧化碳报警的视觉指示。除非支持DeviceMuted属性并将其设置为静音，否则该值应表示报警器当前表示由互连引起的一氧化碳报警的声音指示。

6.3.8.5.4 MuteStateEnum 类型

该数据类型源自enum8，详见下表142。

表142 MuteStateEnum 类型表

| 值 | 名称       | 概述  | 一致性 |
|---|----------|-----|-----|
| 0 | NotMuted | 未静音 | M   |
| 1 | Muted    | 静音  | M   |

其中，

- NotMuted值应表示设备未静音。
- Muted值应表示设备已静音。

6.3.8.5.5 EndOfServiceEnum 类型

该数据类型源自enum8，详见下表143。

表143 EndOfServiceEnum 类型表

| 值 | 名称      | 概述        | 一致性 |
|---|---------|-----------|-----|
| 0 | Normal  | 设备未过期     | M   |
| 1 | Expired | 设备已达到服务期限 | M   |

其中，

- Expired值应表示设备已达到服务期限，需要更换。
- Normal值应表示设备尚未达到服务期限，不需要立即更换。

6.3.8.5.6 ContaminationStateEnum 类型

该数据类型源自enum8，详见下表144。

表144 ContaminationStateEnum 类型表

| 值 | 名称       | 概述           | 一致性 |
|---|----------|--------------|-----|
| 0 | Normal   | 正常状态，传感器未受污染 | M   |
| 1 | Low      | 低污染          | O   |
| 2 | Warning  | 警告状态         | O   |
| 3 | Critical | 临界状态，会发出骚扰警报 | M   |

其中，

- Normal值应表示烟雾传感器具有正常污染级别，客户无需采取任何措施。
- Low值应表示烟雾传感器具有可检测到的污染水平，但污染程度太低，无法发出可见或可闻警报。
- Warning值应表示烟雾传感器的污染水平处于警告状态。在此级别下，污染可能会导致可见或可闻警报。建议用户进行干预。
- Critical值应表示烟雾传感器的污染水平处于临界状态。在此级别，污染应触发可见或可闻警报。需要用户进行干预。传感器的严重污染也应反映为硬件故障（HardwareFault）。

6.3.8.6 属性

该功能集属性详见下表145。

表145 烟雾—氧化碳警报功能集属性表

| 标识符    | 名称                     | 类型                     | 约束条件 | 质量等级 | 默认 | 通道   | 一致性     |
|--------|------------------------|------------------------|------|------|----|------|---------|
| 0x0000 | ExpressedState         | ExpressedStateEnum     | all  | N    | —  | RV   | M       |
| 0x0001 | SmokeState             | AlarmStateEnum         | all  | N    | —  | RV   | SMOKE   |
| 0x0002 | COState                | AlarmStateEnum         | all  | N    | —  | RV   | CO      |
| 0x0003 | BatteryAlert           | AlarmStateEnum         | all  | N    | —  | RV   | M       |
| 0x0004 | DeviceMuted            | MuteStatEnum           | all  | N    | —  | RV   | O       |
| 0x0005 | TestInProgress         | bool                   | all  | —    | —  | RV   | M       |
| 0x0006 | HardwareFaultAlert     | bool                   | all  | N    | —  | RV   | M       |
| 0x0007 | EndOfServiceAlert      | EndOfServiceEnum       | all  | N    | —  | RV   | M       |
| 0x0008 | InterconnectSmokeAlarm | AlarmStateEnum         | all  | —    | —  | RV   | O       |
| 0x0009 | InterconnectCOAlarm    | AlarmStateEnum         | all  | —    | —  | RV   | O       |
| 0x000A | ContaminationState     | ContaminationStateEnum | all  | —    | —  | RV   | [SMOKE] |
| 0x000B | SmokeSensitivityLevel  | SensitivityEnum        | all  | —    | —  | RWVM | [SMOKE] |
| 0x000C | ExpiryDate             | epoch-s                | all  | F    | —  | RV   | O       |

其中，

- ExpressedState属性：该属性应指示报警的可视和可听状态。当服务端中反映多个警报状态时，该属性应显示优先级最高的状态。状态的优先级顺序由制造商决定，应作为认证程序的一部分提供。如果ExpressedState的值不是Normal，则该值对应的属性值不得为Normal。例如，如果ExpressedState设置为SmokeAlarm，则SmokeState的值将表示警报的严重程度（Warning或Critical）。客户端还应读取其他属性，以了解ExpressedState所指示的警报条件之外的其他警报条件。视觉表达通常是LED灯光模式。听觉表达是喇叭或扬声器模式。如果支持DeviceMuted属性并将其设置为静音，则声音表达应被抑制。
- SmokeState属性：该属性应指示设备的烟雾传感器当前是否触发烟雾报警。
- COState属性：该属性应指示设备的一氧化碳传感器当前是否触发一氧化碳报警。
- BatteryAlert属性：该属性应指示设备当前是否触发了电源故障检测机制。如果检测机制被触发，该属性应设置为Warning或Critical，否则应设置为Normal。电池状态也应通过相应的支持属性和事件反映在代表设备电池的电源Cluster中。
- DeviceMuted属性：该属性应指示设备的听觉表达当前是否为静音。听觉表达通常是喇叭或扬声器模式。
- TestInProgress属性：该属性应指示设备自检当前是否已激活。如果设备自检已激活，该属性应设置为True，否则应设置为False。
- HardwareFaultAlert属性：该属性应表明硬件故障检测机制当前是否已触发。如果检测机制已触发，该属性应设置为True，否则应设置为False。
- EndOfServiceAlert属性：该属性应指示设备是否已触发服务终止。当设备达到服务终止状态时，该属性应设置为Expired。
- InterconnectSmokeAlarm属性：该属性应指示互联烟雾报警器当前是否被分支设备触发。当互联烟雾报警器正在被触发时，该属性应设置为Warning或Critical，否则应设置为Normal。
- InterconnectCOAlarm属性：该属性应表明互连的一氧化碳报警器目前是否由分支设备触发。当互连一氧化碳报警器正在被触发时，该属性应设置为Warning或Critical，否则应设置为Normal。
- ContaminationState属性：该属性应指示烟雾传感器的污染级别。
- SmokeSensitivityLevel属性：该属性应指示设备上配置的烟雾传感器的灵敏度级别。
- ExpiryDate属性：该属性应指示设备达到其规定有效期的日期。达到ExpiryDate后，将开始触发EndOfServiceAlert。为了改善跨时区的客户体验，服务终止警报最多可在ExpiryDate之后延迟24小时。同样，客户可以将基于ExpiryDate的任何操作最多延迟24小时，以便与当地时区保持一致。

6.3.8.7 命令

该功能集命令详见下表146。

表146 烟雾一氧化碳警报功能集命令表

| 标识符  | 名称              | 方向            | 返回 | 通道 | 一致性 |
|------|-----------------|---------------|----|----|-----|
| 0x00 | SelfTestRequest | client⇒server | Y  | 0  | 0   |

其中，SelfTestRequest命令应启动设备自检。返回状态应表明测试是否成功启动。一次只能处理一个SelfTestRequest。当ExpressedState属性的值为SmokeAlarm、COAlarm、Testing、InterconnectSmoke、InterconnectCO中的任何一个时，设备不得执行自检，并且应返回状态代码BUSY。

在成功接受SelfTestRequest后，TestInProgress属性应设置为True，ExpressedState属性应设置为Testing。测试过程中发现的任何故障都应反映在相应的属性和事件中。自检程序完成后，应生成SelfTestComplete事件，TestInProgress属性应设置为False，ExpressedState属性应更新以反映服务端的当前状态。

6.3.8.8 事件

6.3.8.8.1 事件列表

该功能集事件详见下表147。

表147 烟雾—氧化碳警报功能集事件表

| 标识符  | 名称                     | 重要等级     | 质量等级 | 通道 | 一致性     |
|------|------------------------|----------|------|----|---------|
| 0x00 | SmokeAlarm             | CRITICAL | —    | V  | SMOKE   |
| 0x01 | COAlarm                | CRITICAL | —    | V  | CO      |
| 0x02 | LowBattery             | INFO     | —    | V  | M       |
| 0x03 | HardwareFault          | INFO     | —    | V  | M       |
| 0x04 | EndOfService           | INFO     | —    | V  | M       |
| 0x05 | SelfTestComplete       | INFO     | —    | V  | M       |
| 0x06 | AlarmMuted             | INFO     | —    | V  | O       |
| 0x07 | MuteEnded              | INFO     | —    | V  | O       |
| 0x08 | InterconnectSmokeAlarm | CRITICAL | —    | V  | [SMOKE] |
| 0x09 | InterconnectCOAlarm    | CRITICAL | —    | V  | [CO]    |
| 0x0A | AllClear               | INFO     | —    | V  | M       |

6.3.8.8.2 SmokeAlarm 事件

当SmokeState属性变为Warning或Critical状态时，应生成该事件。该事件的数据应包含如表148所示的信息。

表148 SmokeAlarm 事件表

| 标识符 | 名称                 | 类型             | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|--------------------|----------------|------|------|----|-----|
| 0   | AlarmSeverityLevel | AlarmStateEnum | all  | —    | —  | M   |

其中，AlarmSeverityLevel字段应指示SmokeState属性的当前值。

6.3.8.8.3 COAlarm 事件

当COState属性变为Warning或Critical状态时，应生成该事件。该事件的数据应包含如表149所示的信息。

表149 COAlarm 事件表

| 标识符 | 名称                 | 类型             | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|--------------------|----------------|------|------|----|-----|
| 0   | AlarmSeverityLevel | AlarmStateEnum | all  | —    | —  | M   |

其中，AlarmSeverityLevel字段应指示COState属性的当前值。

6.3.8.8.4 LowBattery 事件

当BatteryAlert属性变为Warning或Critical状态时，应生成该事件。该事件的数据应包含如表150所示的信息。

表150 LowBattery 事件表

| 标识符 | 名称                 | 类型             | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|--------------------|----------------|------|------|----|-----|
| 0   | AlarmSeverityLevel | AlarmStateEnum | all  | —    | —  | M   |

其中，AlarmSeverityLevel字段应指示BatteryAlert属性的当前值。

6.3.8.8.5 HardwareFault 事件

当设备检测到硬件故障并将HardwareFaultAlert设为True时，应生成该事件。

6.3.8.8.6 EndOfService 事件

当EndOfServiceAlert设置为Expired时，应生成该事件。

6.3.8.8.7 SelfTestComplete 事件

当自检完成且属性TestInProgress变为False时，应生成该事件。

6.3.8.8.8 AlarmMuted 事件

当DeviceMuted属性变为Muted时，应生成该事件。

6.3.8.8.9 MuteEnded 事件

当DeviceMuted属性更改为NotMuted时，应生成该事件。

6.3.8.8.10 InterconnectSmokeAlarm 事件

当服务端所在设备从互联传感器接收到烟雾报警时，应生成该事件。  
该事件的数据应包含的信息详见下表151。

表151 InterconnectSmokeAlarm 事件表

| 标识符 | 名称                 | 类型             | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|--------------------|----------------|------|------|----|-----|
| 0   | AlarmSeverityLevel | AlarmStateEnum | all  | -    | -  | M   |

其中，AlarmSeverityLevel字段应指示InterconnectSmokeAlarm属性的当前值。

6.3.8.8.11 InterconnectCOAlarm 事件

当服务端所在设备收到来自互联传感器的一氧化碳警报时，应生成该事件。该事件的数据应包含如表152所示的信息。

表152 InterconnectCOAlarm 事件表

| 标识符 | 名称                 | 类型             | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|--------------------|----------------|------|------|----|-----|
| 0   | AlarmSeverityLevel | AlarmStateEnum | all  | -    | -  | M   |

其中，AlarmSeverityLevel字段应指示InterconnectCOAlarm属性的当前值。

6.3.8.8.12 AllClear 事件

当ExpressedState属性返回Normal状态时，应生成该事件。

6.3.9 电能测量功能集

6.3.9.1 概述

此功能集提供了一种查询服务端输入或输出的电能数据的机制。

6.3.9.2 分类

该功能集分类详见下表153。

表153 电能测量功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | EEM     |

6.3.9.3 Cluster ID

该功能集ID详见下表154。

表154 电能测量功能集 ID 表

| 标识符    | 名称                          |
|--------|-----------------------------|
| 0x0091 | ElectricalEnergyMeasurement |

### 6.3.9.4 特性

该功能集应支持如下定义的FeatureMapbitmap属性，详见下表155。

表155 电能测量功能集位图属性表

| Bit | 编码   | 特征               | 一致性   | 概述         |
|-----|------|------------------|-------|------------|
| 0   | IMPE | ImportedEnergy   | 0. a+ | 测量服务端输入的能量 |
| 1   | EXPE | ExportedEnergy   | 0. a+ | 测量服务端输出的能量 |
| 2   | CUME | CumulativeEnergy | 0. b+ | 累积测量值      |
| 3   | PERE | PeriodicEnergy   | 0. b+ | 周期测量值      |

其中，

- 输入能量（ImportedEnergy）特性表示服务端能够测量服务端输入了多少能量。
- 输出能量（ExportedEnergy）特性表明服务端能够测量服务端输出了多少能量。
- 累计能量（CumulativeEnergy）特性表明服务端能够测量在设备的生命周期内服务端输入或输出了多少能量。该测量可以从设备固件更新到包含该功能时、设备固件更新到纠正测量错误时、或者设备恢复出厂时开始。
- 周期能量（PeriodicEnergy）特性表明服务端能够测量在一段时间内服务端输入或输出了多少能量。测量周期的开始和结束时间应由服务端确定，并且可以代表重合的周期。

### 6.3.9.5 数据类型

#### 6.3.9.5.1 EnergyMeasurementStruct 类型

该结构应指示在给定测量周期内测量的能量值。

无法确定UTC时间或尚未确定UTC时间的服务端，应使用系统时间字段来指定测量周期和观测时间。

已确定UTC时间的服务端应使用时间戳字段来指定测量周期。该服务端还可以包括systime字段，以指示从给定的时间戳启动以来已经过了多少秒；这允许客户端解析以前仅包含systime的报告的UTC时间，详见下表156。

表156 EnergyMeasurementStruct 类型表

| 标识符 | 名称             | 类型         | 约束条件                    | 质量等级 | 默认 | 通道 | 一致性  |
|-----|----------------|------------|-------------------------|------|----|----|------|
| 0   | Energy         | energy-mWh | 0to262                  | —    | —  | —  | M    |
| 1   | StartTimestamp | epoch-s    |                         | —    | —  | —  | desc |
| 2   | EndTimestamp   | epoch-s    | min<br>(StartSystime+1) | —    | —  | —  | desc |
| 3   | StartSystime   | systime-ms |                         | —    | —  | —  | desc |
| 4   | EndSystime     | systime-ms | min<br>(StartSystime+1) | —    | —  | —  | desc |

其中，

——Energy 字段

该字段应为报告的能量。

如果EnergyMeasurementStruct表示累积能量，则这应表示在EndTimestamp字段值或EndSystime字段值记录的累积能量，或两者都记录。

如果EnergyMeasurementStruct表示周期能量，则这应表示在StartTimestamp和EndTimestamp字段指定的时间段内，或在StartSystime和EndSystime字段指定的时间段内记录的能量，或两者都记录。

——StartTimestamp 字段

该字段应指示测量Energy字段值的周期开始的时间戳（UTC）。

如果该EnergyMeasurementStruct表示累计能量，则应省略该字段。

否则，如果服务端在测量周期开始时或开始前确定了UTC时间，则应指示该字段。

否则，如果服务端在测量期间开始时或开始前尚未确定UTC时间，或者无法确定UTC时间，则应省略该字段。

——EndTimestamp 字段

该字段应指示测量Energy字段值的周期结束的时间戳（UTC）。

如果服务端在测量周期结束时已经确定了UTC时间，则应指示该字段。

否则，如果服务端在测量周期结束时尚未确定UTC时间，或者无法确定UTC时间，则应省略该字段。

——StartSystime 字段

该字段应指示在从启动后，至测量Energy字段值的周期开始时所经过的时间。如果该EnergyMeasurementStruct表示累计电能，则应省略该字段。

否则，如果服务端在测量周期开始时尚未确定UTC时间，或者无法确定UTC时间，则应指示该字段。

否则，如果服务端在测量周期开始时或开始之前确定了UTC时间，则可以省略该字段；如果有指示，其值应为从启动后至StartTimestamp中指示的UTC时间所经过的时间。

——EndSystime 字段

该字段应指示在从启动后，至测量Energy字段值的周期结束时所经过的时间。

如果服务端在测量周期结束时尚未确定UTC时间，或者无法确定UTC时间，则应指示该字段。

否则，如果服务端在测量周期结束时已经确定了UTC时间，则可以省略该字段；如果有指示，其值应为从启动后至EndTimestamp中指示的UTC时间所经过的时间。

6.3.9.5.2 CumulativeEnergyResetStruct 类型

此结构应表示累积测量值最后一次归零的时间，该时间可能是由于设备初始化或累积值的内部复位，详见下表157。

表157 CumulativeEnergyResetStruct 类型表

| 标识符 | 名称                     | 类型         | 约束条件 | 质量等级 | 默认   | 通道 | 一致性    |
|-----|------------------------|------------|------|------|------|----|--------|
| 0   | ImportedResetTimestamp | epoch-s    | –    | X    | null | –  | [IMPE] |
| 1   | ExportedResetTimestamp | epoch-s    | –    | X    | null | –  | [EXPE] |
| 2   | ImportedResetSystime   | systime-ms | –    | X    | null | –  | [IMPE] |
| 3   | ExportedResetSystime   | systime-ms | –    | X    | null | –  | [EXPE] |

其中，

——ImportedResetTimestamp 字段

这个字段应指示UTC时间戳，即当CumulativeEnergyImported属性中的Energy字段的值最近一次为零时的时间。

如果服务端已经确定了CumulativeEnergyImported属性的Energy字段的值最近为零时的UTC时间，则应指示该字段。

否则，如果服务端尚未确定CumulativeEnergyImported属性的Energy字段的值最近为零时的UTC时间，或者无法确定UTC时间，则应忽略该字段。

如果当前无法确定当CumulativeEnergyImported属性中的Energy字段的值最近一次为零时的UTC时间戳，则应返回一个null值。

——ExportedResetTimestamp 字段

这个字段应指示UTC时间戳，即当CumulativeEnergyExported属性中的Energy字段的值最近一次为零时的时间。

如果服务端已经确定了CumulativeEnergyExported属性的Energy字段的值最近为零时的UTC时间，则应指示该字段。

否则，如果服务端尚未确定CumulativeEnergyExported属性的Energy字段的值最近为零时的UTC时间，或者无法确定UTC时间，则应忽略该字段。

如果当前无法确定CumulativeEnergyExported属性的Energy字段值最近为零时的UTC时间戳，则应返回null值。

——ImportedResetSystime 字段

此字段应指示自启动以来到CumulativeEnergyImported属性的Energy字段值最近一次为零时所经过的时间

如果服务端尚未确定CumulativeEnergyImported属性的Energy字段的值最近为零时的UTC时间，或者无法确定UTC时间，则应指示该字段。

否则，如果服务端已经确定了CumulativeEnergyImported属性的Energy字段的值最近为零时的UTC时间，则可以省略该字段。如果有指示，其值应为从启动后至ImportedResetTimestamp中指示的UTC时间所经过的时间。

——ExportedResetSystime 字段

此字段应指示自启动以来到CumulativeEnergyExported属性的Energy字段值最近一次为零时所经过的时间。如果服务端尚未确定CumulativeEnergyExported属性的Energy字段的值最近为零时的UTC时间，或者无法确定UTC时间，则应指示该字段。

否则，如果服务端已经确定了CumulativeEnergyExported属性的Energy字段的值最近为零时的UTC时间，则可以省略该字段。如果有指示，其值应为从启动后至ImportedResetTimestamp中指示的UTC时间所经过的时间。

6.3.9.6 属性

该功能集属性详见下表158。

表158 电能测量功能集属性表

| 标识符    | 名称                       | 类型                          | 约束条件 | 质量等级 | 默认   | 通道 | 一致性       |
|--------|--------------------------|-----------------------------|------|------|------|----|-----------|
| 0x0000 | Accuracy                 | MeasurementAccuracyStruct   | —    | F    | —    | RV | M         |
| 0x0001 | CumulativeEnergyImported | EnergyMeasurementStruct     | —    | XQ   | null | RV | IMPE&CUME |
| 0x0002 | CumulativeEnergyExported | EnergyMeasurementStruct     | —    | XQ   | null | RV | EXPE&CUME |
| 0x0003 | PeriodicEnergyImported   | EnergyMeasurementStruct     | —    | XQ   | null | RV | IMPE&PERE |
| 0x0004 | PeriodicEnergyExported   | EnergyMeasurementStruct     | —    | XQ   | null | RV | EXPE&PERE |
| 0x0005 | CumulativeEnergyReset    | CumulativeEnergyResetStruct | —    | X    | null | RV | [CUME]    |

其中，

——Accuracy 属性

该属性应表明该服务端能量测量的准确性。MeasurementAccuracyStruct上的MeasurementType字段的值应为电能。

——CumulativeEnergyImported 属性

该属性应指示服务端在设备生命期间内输入的累积能量的最新测量值，以及测量值记录时的时间戳。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之内，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将此属性标记为报告就绪的时间，但服务端不得将此属性标记为报告就绪的时间延迟超过60秒。

如果当前无法确定输入的累积能量，则应返回null值。

——CumulativeEnergyExported 属性

此属性应指示服务端在设备生命周期内输出的累积能量的最新测量值，以及测量值记录时的时间戳。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之内，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将此属性标记为报告就绪的时间，但服务端不得将此属性标记为报告就绪的时间延迟超过60秒。

如果当前无法确定输出的累积能量，则应返回null值。

——PeriodicEnergyImported 属性

此属性应指示服务端在能量测量周期内输入能量的最新测量值。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之内，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将此属性标记为报告就绪的时间，但服务端不得将此属性标记为报告就绪的时间延迟超过60秒。

如果当前无法确定输入的周期能量，则应返回null值。

——PeriodicEnergyExported 属性

此属性应指示服务端在能量测量周期内输出能量的最新测量值。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之内，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将此属性标记为报告就绪的时间，但服务端不得将此属性标记为报告就绪的时间延迟超过60秒。

如果当前无法确定输出的周期能量，则应返回null值。

——CumulativeEnergyReset 属性

此属性应指示最近累积测量值为零的时间。

6.3.9.7 事件

此功能集应支持的事件详见下表159。

表159 电能测量功能集事件表

| 标识符 | 名称                       | 重要等级 | 质量等级 | 通道 | 一致性  |
|-----|--------------------------|------|------|----|------|
| 0   | CumulativeEnergyMeasured | INFO | —    | V  | CUME |
| 1   | PeriodicEnergyMeasured   | INFO | —    | V  | PERE |

其中，

——CumulativeEnergyMeasured 事件

当服务端对服务端输入的累积能量、输出的累积能量或两者同时进行快照时，将产生此事件，但不会比上述相关属性的描述中提到的频率更频繁。

此事件的数据应包含如表160所示的信息。

表160 CumulativeEnergyMeasured 事件表

| 标识符 | 名称             | 类型                      | 约束条件 | 质量等级 | 默认 | 一致性       |
|-----|----------------|-------------------------|------|------|----|-----------|
| 0   | EnergyImported | EnergyMeasurementStruct | —    | —    | —  | CUME&IMPE |
| 1   | EnergyExported | EnergyMeasurementStruct | —    | —    | —  | CUME&EXPE |

其中，

- EnergyImported字段应为在EndTimestamp字段,EndSystime字段或两者所指示的时间戳时的CumulativeEnergyImported属性的值。
- EnergyExported字段应为在EndTimestamp字段,EndSystime字段或两者所指示的时间戳时的CumulativeEnergyExported属性的值。

——PeriodicEnergyMeasured 事件

当服务端达到输入能量、输出能量，或两者的报告周期结束点时，将生成此事件。此事件的数据应包含如表161所示的信息。

表161 PeriodicEnergyMeasured 事件表

| 标识符 | 名称             | 类型                      | 约束条件 | 质量等级 | 默认 | 一致性       |
|-----|----------------|-------------------------|------|------|----|-----------|
| 0   | EnergyImported | EnergyMeasurementStruct | —    | —    | —  | PERE&IMPE |
| 1   | EnergyExported | EnergyMeasurementStruct | —    | —    | —  | PERE&EXPE |

其中，

- EnergyImported字段应为在EndTimestamp字段,EndSystime字段或两者所指示的时间戳时的PeriodicEnergyImported属性的值。

- EnergyExported字段应为在EndTimestamp字段,EndSystime字段或两者所指示的时间戳时的PeriodicEnergyExported属性的值。

### 6.3.10 电力测量功能集

#### 6.3.10.1 概述

此功能集提供了一种用于查询服务端测量的电力数据的机制。

#### 6.3.10.2 分类

该功能集分类详见下表162。

表162 电力测量功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | EPM     |

#### 6.3.10.3 Cluster ID

该功能集ID详见下表163。

表163 电力测量功能集 ID 表

| 标识符    | 名称                         |
|--------|----------------------------|
| 0x0090 | ElectricalPowerMeasurement |

#### 6.3.10.4 特性

该功能集应支持如下定义的FeatureMapbitmap属性，详见下表164。

表164 电力测量功能集位图属性表

| Bit | 编码   | 特征                 | 一致性    | 概述          |
|-----|------|--------------------|--------|-------------|
| 0   | DIRC | DirectCurrent      | 0. a+  | 支持直流电测量     |
| 1   | ALTC | AlternatingCurrent | 0. a+  | 支持交流电测量     |
| 2   | POLY | PolyphasePower     | [ALTC] | 支持多相测量      |
| 3   | HARM | Harmonics          | [ALTC] | 支持交流谐波测量    |
| 4   | PWRQ | PowerQuality       | [ALTC] | 支持交流谐波相位的测量 |

其中，

- DirectCurrent特性表明组合仪表可以测量直流电。
- AlternatingCurrent特性表明组合仪表可以测量交流电。
- PolyphasePower特性表明该功能集代表多相电源的集合测量值。
- Harmonics特性表明组合仪表可以测量交流电的谐波。
- PowerQuality特性表明组合仪表可以测量交流电的谐波相位。

#### 6.3.10.5 数据类型

##### 6.3.10.5.1 PowerModeEnum 类型

该数据类型源自enum8，详见下表165。

表165 PowerModeEnum 类型表

| 值 | 名称      | 概述       | 一致性 |
|---|---------|----------|-----|
| 0 | Unknown | -        | M   |
| 1 | DC      | 直流电      | M   |
| 2 | AC      | 单相或多相交流电 | M   |

##### 6.3.10.5.2 MeasurementRangeStruct 类型

此结构应指明在一个测量周期内给定测量类型的最大值和最小值，以及这些值的观测次数。

无法确定UTC时间或尚未确定UTC时间的服务端，应使用系统时间字段来指定测量周期和观察时间。

已确定UTC时间的服务端应使用时间戳字段来指定测量周期和观测次数。这种服务端还可以包括 systime 字段，以指示从启动后至给定的时间戳已经过了多少秒；这允许客户端解析以前仅包含 systime 的报告的UTC时间。详见下表166。

表166 MeasurementRangeStruct 类型表

| 标识符 | 名称              | 类型                  | 约束条件                      | 质量等级 | 默认 | 通道 | 一致性          |
|-----|-----------------|---------------------|---------------------------|------|----|----|--------------|
| 0   | MeasurementType | MeasurementTypeEnum | —                         | —    | —  | —  | M            |
| 1   | Min             | int64               | -262to262                 | —    | —  | —  | M            |
| 2   | Max             | int64               | -262to262                 | —    | —  | —  | M            |
| 3   | StartTimestamp  | epoch-s             |                           | —    | —  | —  | EndTimestamp |
| 4   | EndTimestamp    | epoch-s             | min<br>(StartTimestamp+1) | —    | —  | —  | desc         |
| 5   | MinTimestamp    | epoch-s             | —                         | —    | —  | —  | EndTimestamp |
| 6   | MaxTimestamp    | epoch-s             | min<br>(MinTimestamp+1)   | —    | —  | —  | EndTimestamp |
| 7   | StartSystime    | systime-ms          | —                         | —    | —  | —  | EndSystime   |
| 8   | EndSystime      | systime-ms          | min<br>(StartSystime+1)   | —    | —  | —  | desc         |
| 9   | MinSystime      | systime-ms          | —                         | —    | —  | —  | EndSystime   |
| 10  | MaxSystime      | systime-ms          | min (MinSystime+1)        | —    | —  | —  | EndSystime   |

其中，

——MeasurementType 字段

该字段应为所提供范围的测量类型。

——Min 字段

该字段应为StartTimestamp和EndTimestamp之间，或StartSystime和EndSystime之间，或两者之间的最小测量值。

——Max 字段

该字段应为StartTimestamp和EndTimestamp之间，或StartSystime和EndSystime之间，或两者之间的最大测量值。

——StartTimestamp 字段

该字段应为测量周期开始时的UTC时间戳。

如果服务端在测量周期开始时或开始前尚未确定UTC时间，或者无法确定UTC时间，则应省略该字段。

——EndTimestamp 字段

该字段应为测量周期结束时的UTC时间戳。

如果服务端在测量期间开始时或开始前尚未确定UTC时间，或者无法确定UTC时间，则应省略此字段。

——MinTimestamp 字段

该字段应为测量Min字段值的最新UTC时间戳。该字段应大于或等于StartTimestamp字段的值。该字段应小于或等于EndTimestamp字段的值。

——MaxTimestamp 字段

该字段应为测量Max字段值的最新UTC时间戳。该字段应大于或等于StartTimestamp字段的值。该字段应小于或等于EndTimestamp字段的值。

——StartSystime 字段

该字段应为从启动至测量周期开始的时间。

如果服务端在测量周期开始时或开始前确定了UTC时间，则可以省略此字段以及EndSystime、MinSystime和MaxSystime字段。

——EndSystime 字段

该字段应为从启动至测量周期结束的时间。

如果服务端在测量周期结束时确定了UTC时间，则可以省略此字段以及StartSystime、MinSystime和MaxSystime字段。

——MinSystime 字段

该字段应为自测量Min字段中的值启动以来的测量时间。该字段应大于或等于StartSystime字段的值。

该字段应小于或等于EndSystime字段的值。

——MaxSystime 字段

该字段应为自测量Max字段中的值启动以来的测量时间。该字段应大于或等于StartSystime字段的值。

该字段应小于或等于EndSystime字段的值。

6.3.10.5.3 HarmonicMeasurementStruct 类型

HarmonicMeasurementStruct类型详见下表167。

表167 HarmonicMeasurementStruct 类型表

| 标识符 | 名称          | 类型    | 约束条件      | 质量等级 | 默认   | 通道 | 一致性 |
|-----|-------------|-------|-----------|------|------|----|-----|
| 0   | Order       | uint8 | min1      | -    | 1    | -  | M   |
| 1   | Measurement | int64 | -262to262 | X    | null | -  | M   |

其中，

——Order 字段

该字段应为被测谐波的阶数。通常这是一个奇数，但是服务端可以选择报告偶次谐波。

——Measurement 字段

该字段应为给定谐波阶次的测量值。

对于HarmonicCurrents属性，该值是最新测量的谐波电流读数，单位为毫安（mA）。正值表示测得的谐波电流为正，负值表示测得的谐波电流为负。

对于HarmonicPhases属性，该值是给定谐波阶数的最新相位，单位为毫度（mDeg）。正值表示测量相位超前，负值表示测量相位滞后。

如果该测量当前不可用，则应返回空值。

6.3.10.6 属性

该功能集属性详见下表168。

表168 电力测量功能集属性表

| ID     | 名称                       | 类型                                     | 约束条件                        | 质量等级 | 默认    | 通道 | 一致性 |
|--------|--------------------------|----------------------------------------|-----------------------------|------|-------|----|-----|
| 0x0000 | PowerMode                | PowerModeEnum                          | -                           | -    | -     | RV | M   |
| 0x0001 | NumberOfMeasurementTypes | uint8                                  | min1                        | F    | -     | RV | M   |
| 0x0002 | Accuracy                 | list[章节“MeasurementAccuracyStruct 类型”] | 1toNumberOfMeasurementTypes | F    | -     | RV | M   |
| 0x0003 | Ranges                   | list[MeasurementRangeStruct]           | 0toNumberOfMeasurementTypes | Q    | empty | RV | O   |
| 0x0004 | Voltage                  | voltage-mV                             | -262to262                   | XQ   | null  | RV | O   |
| 0x0005 | ActiveCurrent            | amperage-mA                            | -262to262                   | XQ   | null  | RV | O   |

表168 电力测量功能集属性表（续）

| ID     | 名称               | 类型                              | 约束条件              | 质量等级 | 默认   | 通道 | 一致性    |
|--------|------------------|---------------------------------|-------------------|------|------|----|--------|
| 0x0006 | ReactiveCurrent  | amperage-mA                     | -262to262         | XQ   | null | RV | [ALTC] |
| 0x0007 | ApparentCurrent  | amperage-mA                     | 0to262            | XQ   | null | RV | [ALTC] |
| 0x0008 | ActivePower      | power-mW                        | -262to262         | XQ   | null | RV | M      |
| 0x0009 | ReactivePower    | power-mW                        | -262to262         | XQ   | null | RV | [ALTC] |
| 0x000A | ApparentPower    | power-mW                        | -262to262         | XQ   | null | RV | [ALTC] |
| 0x000B | RMSVoltage       | voltage-mV                      | -262to262         | XQ   | null | RV | [ALTC] |
| 0x000C | RMSCurrent       | amperage-mA                     | -262to262         | XQ   | null | RV | [ALTC] |
| 0x000D | RMSPower         | power-mW                        | -262to262         | XQ   | null | RV | [ALTC] |
| 0x000E | Frequency        | int64                           | 0to<br>1000000    | XQ   | null | RV | [ALTC] |
| 0x000F | HarmonicCurrents | list[HarmonicMeasurementStruct] | desc              | XQ   | null | RV | HARM   |
| 0x0010 | HarmonicPhases   | list[HarmonicMeasurementStruct] | desc              | XQ   | null | RV | PWRQ   |
| 0x0011 | PowerFactor      | int64                           | -10000to<br>10000 | XQ   | null | RV | [ALTC] |
| 0x0012 | NeutralCurrent   | amperage-mA                     | -262to262         | XQ   | null | RV | [POLY] |

其中，

——PowerMode 属性

这将显示服务端的当前模式。对于某些服务端，比如EV，这可能会因充电或放电模式而异。

——NumberOfMeasurementTypes 属性

这将指示服务端能够报告的测量类型的最大数量。

——Accuracy 属性

这将显示服务端支持的精度规范的测量类型列表。应有一项ActivePower，以及该服务端实现的任何其他测量类型。

——Ranges 属性

这将显示不同测量类型的测量范围列表。每个测量类型在此列表中最多有一项，代表最近测量周期内的测量范围。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

——Voltage 属性

这将显示最新的电压读数，单位为毫伏（mV）。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果无法测量电压，则应返回null值。

——ActiveCurrent 属性

这将显示最新的有功电流读数，单位为毫安（mA）。

正值表示流入服务端的电流，负值表示流出服务端的电流。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果无法测量电流，则应返回null值。

——ReactiveCurrent 属性

这将显示最新的无功电流读数，单位为毫安（mA）。

正值表示流入服务端的电流，负值表示流出服务端的电流。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果无法测量电流，则应返回null值。

——ApparentCurrent 属性

这将显示最近的表现电流（有功和无功电流平方的平方根和）读数，单位为毫安（mA）。该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。

如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果无法测量有功或无功电流，则应返回null值。

——ActivePower 属性

这将显示最新的有功功率读数，单位为毫瓦（mW）。如果无法测量功率，则应返回null值。

正值表示输入功率，负值表示输出功率。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果设置了多相功率特性，该值代表输入或输出的总有功功率。

——ReactivePower 属性

这将显示最新的无功功率读数，单位为无功毫伏安培（mVAR）。

正值表示输入功率，负值表示输出功率。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果无功功率无法测量，则应返回null值。

如果支持多相功率特性，该值表示输入或输出的总无功功率。

——ApparentPower 属性

这将显示最新的表现功率读数，单位为毫伏安培（mVA）。

正值表示输入功率，负值表示输出功率。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果无法测量表现功率，则应返回null值。

——RMSVoltage 属性

这将显示最新的RMS电压读数，单位为毫伏（mV）。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果无法测量均方根电压，则应返回null值。

——RMSCurrent 属性

这将显示最新的RMS电流读数，单位为毫安（mA）。

正值表示流入服务端的电流，负值表示流出服务端的电流。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果无法测量均方根电流，则应返回null值。

——RMSPower 属性

这将显示最新的RMS功率读数，单位为毫瓦（mW）。

正值表示输入功率，负值表示输出功率。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果无法测量均方根功率，则应返回null值。

——Frequency 属性

这将显示最新的频率读数，单位为毫赫兹（mHz）。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

如果无法测量频率，则应返回null值。

——HarmonicCurrents 属性

这将显示谐波测量结构值列表，每个谐波测量结构代表阶数规定的谐波订单的谐波电流读数。该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。

如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

——HarmonicPhases 属性

这将显示谐波测量结构值列表，每个谐波测量结构代表阶数指定谐波订单的谐波电流读数的最新相位。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

——PowerFactor 属性

这应以+/-1/100%的百分比表示功率因数比。

该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

——NeutralCurrent 属性

这将显示最近的中性电流读数，单位为毫安（mA）。通常，这是一个输出值，采用相电流矢量和的幅度。

如果零线电流无法测量或导出，则应返回null值。

正值表示输入电源时相电流之间的不平衡。负值表示输出功率时相电流之间的不平衡。该属性的报告间隔应取决于制造商。服务端可以选择省略发布被认为没有意义的增量。如果上一次执行此操作的时间是在1秒之前，则服务端不应将此属性标记为报告就绪。

如果需要，服务端可以延长将时间将此属性标记为报告就绪，但服务端不得延迟将此属性标记为报告就绪超过60秒。

6.3.10.7 事件

此功能集应支持的事件详见下表169。

表169 电力测量功能集事件表

| 标识符 | 名称                      | 重要等级 | 通道 | 一致性    |
|-----|-------------------------|------|----|--------|
| 0   | MeasurementPeriodRanges | INFO | RV | Ranges |

——MeasurementPeriodRanges 事件

如果支持，该事件应在测量周期结束时生成。测量周期的开始和结束时间应由服务端确定，并且可能代表重叠的周期。该事件的数据应包含如表170所示的信息。

表170 MeasurementPeriodRanges 事件表

| 标识符 | 名称     | 类型                           | 约束条件 | 质量等级 | 默认 | 通道 | 一致性 |
|-----|--------|------------------------------|------|------|----|----|-----|
| 0   | Ranges | list[MeasurementRangeStruct] | —    | —    | —  | RV | M   |

其中，Ranges字段将指示事件生成时Ranges属性的值。

6.4 照明

6.4.1 概述

功能集由诸如本章之类的各个章节组成。请参阅功能集中的文档控制以获取所有章节和文档的列表。章节之间的引用使用X.Y表示法，其中X是章节，Y是该章节中的子节。对外部文档的引用包含在参见“2 规范性引用文件”中，并使用[Rn]表示法。

6.4.2 照明功能集列表

6.4.2.1 概述

本文档中指定的功能集通常用于照明应用，但也可以用于任何应用领域。

6.4.2.2 条款

镇流器系数：镇流器和灯组合的光输出（流明）的测量值，与使用相同灯的ANSI标准镇流器进行比较。将镇流器系数乘以灯的额定流明，即可得到灯/镇流器组合的光输出。

HSV：色调、饱和度、明度。一种颜色空间，也称为HSB（色调、饱和度、亮度）。这是RGB（红、绿、蓝）颜色空间的著名变换。有关更多信息，请参阅[http://en.wikipedia.org/wiki/HSV\\_color\\_space](http://en.wikipedia.org/wiki/HSV_color_space)。

照度：表面上的入射光通量密度。照度是照明水平的标准度量，以勒克斯（lx）为单位。

6.4.2.3 功能集列表

本节列出了本章中指定的特定照明功能集，详见下表171。

表171 照明功能集概览

| 功能集标识符 | 功能集名称                | 描述               |
|--------|----------------------|------------------|
| 0x0300 | ColorControl         | 用于控制彩色灯光颜色的属性和命令 |
| 0x0301 | BallastConfiguration | 配置照明镇流器的属性和命令    |

6.4.3 颜色控制功能集

6.4.3.1 概述

此功能集提供更改灯光颜色的接口。颜色根据CIE1931颜色空间指定。颜色控制根据x、y值进行，如本文件所定义。

另外，颜色可以选择性地通过色温、色调和饱和度来控制。基于可选变量RGB和W颜色点的配比值。建议根据HSV（又称HSB）颜色模型来解释色调和饱和度。

不包括亮度控制，因为这是通过照明级别控制功能集提供的。建议将此功能集提供的级别解释为当前颜色可实现的最大强度的比例。

6.4.3.2 分类

标识功能集分类详见下表172。

表172 标识功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | CC      |

6.4.3.3 Cluster ID

标识功能集ID详见下表173。

表173 标识功能集 ID 表

| 标识符    | 名称           |
|--------|--------------|
| 0x0300 | ColorControl |

6.4.3.4 特性

该功能集应支持如下表定义的Feature Map位图属性，详见下表174。

表174 Feature Map 位图属性

| Bit | 编码   | 特征               | 一致性     | 概述             |
|-----|------|------------------|---------|----------------|
| 0   | HS   | HueSaturation    | EHUE, 0 | 支持通过色调/饱和度指定颜色 |
| 1   | EHUE | EnhancedHue      | CL, 0   | 支持增强色调         |
| 2   | CL   | ColorLoop        | 0       | 支持色彩循环         |
| 3   | xy   | xy               | 0       | 支持通过xy指定颜色     |
| 4   | CT   | ColorTemperature | 0       | 支持色温指定         |

6.4.3.5 依赖项

6.4.3.5.1 将色温与电平控制相结合

如果照明级别控制功能集标识符0x0008在与色彩控制功能集相同的端点上受支持，并且支持色温，则可以将当前级别的变化与色温耦合。

级别控制群集的选项属性的CoupleColorTempToLevel位指示色温是否要与级别控制群集中的CurrentLevel属性相链接。

如果级别控制群集的选项属性的CoupleColorTempToLevel位等于1，并且ColorMode或EnhancedColorMode属性设置为2（ColorTemperatureMireds），则CurrentLevel属性的更改将影响ColorTemperatureMireds属性。此关系是制造商特定的，条件是CurrentLevel属性的最大值应当对应等于CoupleColorTempToLevelMinMireds属性的ColorTemperatureMired属性值。此关系是单向的，因此对ColorTemperatureMireds属性的更改将不会对CurrentLevel属性产生任何影响。

为了模拟白炽灯的行为，CurrentLevel属性的低值应与ColorTemperatureMireds属性的高值相关联（即开尔文色温的低值）。

如果级别控制群集的选项属性的CoupleColorTempToLevel位等于0，则色温和当前级别之间不得存在任何联系。

6.4.3.5.2 色调和饱和度的独立过渡

该群集中的各种命令可用于启动色调和/或饱和度的转换。

- 当色调正在转变时，改变饱和度的命令（MoveSaturation）（使用MoveMode!=Stop）、StepSaturation、MoveToSaturation）被服务端接收到，后一个命令不得中断正在进行的色调转换。

- 当饱和度转换正在进行时,服务端收到改变色调的命令(MoveHue)(使用MoveMode!=Stop)、EnhancedMoveHue (使用MoveMode!=Stop) StepHue、EnhancedStepHue、MoveToHue、EnhancedMoveToHue), 后一个命令不得中断正在进行的饱和度转换。

6.4.3.6 数据类型

6.4.3.6.1 颜色能力位图

该数据类型源自map16, 详见下表175。

表175 颜色能力位图表

| Bit | 名称               | 概述               | 一致性  |
|-----|------------------|------------------|------|
| 0   | HueSaturation    | 支持通过色调/饱和度指定颜色变化 | HS   |
| 1   | EnhancedHue      | 支持增强色彩           | EHUE |
| 2   | ColorLoop        | 支持色彩循环           | CL   |
| 3   | xy               | 支持通过xy指定颜色       | xy   |
| 4   | ColorTemperature | 支持通过色温指定颜色       | CT   |

6.4.3.6.2 OptionsBitmap 类型

该数据类型源自map8, 详见下表176。

表176 OptionsBitmap 类型

| Bit | 名称           | 概述         | 一致性 |
|-----|--------------|------------|-----|
| 0   | ExecuteIfOff | 依赖On/Off集群 | M   |

其中, ExecuteIfOff位应指示此功能集服务端实例是否与On/Off功能集具有依赖关系。

6.4.3.6.3 UpdateFlagsBitmap 类型

该数据类型源自map8并用于ColorLoopSet命令, 详见下表177。

表177 UpdateFlagsBitmap 类型

| Bit | 名称              | 概述            | 一致性 |
|-----|-----------------|---------------|-----|
| 0   | UpdateAction    | 设备遵守相关的操作字段   | M   |
| 1   | UpdateDirection | 设备更新相关的方向属性   | M   |
| 2   | UpdateTime      | 设备更新相关的时间属性   | M   |
| 3   | UpdateStartHue  | 设备更新相关的开始色调属性 | M   |

其中,

——UpdateAction 位

该位应指示服务端是否遵守操作字段来处理命令。

0=设备应忽略操作字段。

1=设备应遵守操作字段。

——UpdateDirection 位

该位应指示设备是否使用Direction字段更新ColorLoopDirection属性。

0=设备应忽略方向字段。

1=设备应使用Direction字段的值更新ColorLoopDirection属性。

——UpdateTime 位

此位应指示设备是否使用时间字段更新ColorLoopTime属性。

0=设备应忽略时间字段。

1=设备应使用时间字段的值更新ColorLoopTime属性的值。

——UpdateStartHue 位

该位应指示设备是否使用StartHue字段的值更新ColorLoopStartEnhancedHue属性。

0=设备应忽略StartHue字段。  
1=设备应使用StartHue字段的值更新ColorLoopStartEnhancedHue属性的值。

6.4.3.6.4 DriftCompensationEnum 类型

该数据类型源自enum8，详见下表178。

表178 DriftCompensationEnum 类型

| 值 | 名称                                    | 概述            | 一致性 |
|---|---------------------------------------|---------------|-----|
| 0 | None                                  | 没有补偿          | M   |
| 1 | OtherOrUnknown                        | 补偿基于其他或未知的机制  | M   |
| 2 | TemperatureMonitoring                 | 补偿基于温度监测      | M   |
| 3 | OpticalLuminanceMonitoringAndFeedback | 补偿基于光学亮度监测和反馈 | M   |
| 4 | OpticalColorMonitoringAndFeedback     | 补偿基于光学色彩监控和反馈 | M   |

6.4.3.6.5 ColorModeEnum 类型

该数据类型源自enum8，详见下表179。

表179 Color ModeEnum 类型

| 值 | 名称                             | 概述             | 一致性 |
|---|--------------------------------|----------------|-----|
| 0 | CurrentHueAndCurrentSaturation | 当前色调和饱和度属性决定颜色 | M   |
| 1 | CurrentXAndCurrentY            | 当前X和Y属性决定颜色    | M   |
| 2 | ColorTemperatureMireds         | 色温属性决定颜色       | M   |

6.4.3.6.6 EnhancedColorModeEnum 类型

该数据类型源自enum8，详见下表180。

表180 EnhancedColor ModeEnum 类型

| 值 | 名称                                     | 概述                 | 一致性 |
|---|----------------------------------------|--------------------|-----|
| 0 | CurrentHueAndCurrentSaturation         | 当前色调和饱和度属性决定颜色     | M   |
| 1 | CurrentXAndCurrentY                    | 当前的X和Y属性决定颜色       | M   |
| 2 | ColorTemperatureMireds                 | 色温属性决定颜色           | M   |
| 3 | EnhancedCurrentHueAndCurrentSaturation | 增强的当前色调和饱和度属性决定了颜色 | M   |

6.4.3.6.7 DirectionEnum 类型

该数据类型源自enum8，详见下表181。

表181 DirectionEnum 类型

| 值 | 名称       | 概述   | 一致性 |
|---|----------|------|-----|
| 0 | Shortest | 最短距离 | M   |
| 1 | Longest  | 最长距离 | M   |
| 2 | Up       | 向上   | M   |
| 3 | Down     | 向下   | M   |

6.4.3.6.8 MoveModeEnum 类型

该数据类型源自enum8，详见下表182。

表182 MoveModeEnum 类型

| 值 | 名称   | 概述   | 一致性 |
|---|------|------|-----|
| 0 | Stop | 停止运动 | M   |
| 1 | Up   | 向上移动 | M   |
| 3 | Down | 向下移动 | M   |

#### 6.4.3.6.9 StepModeEnum 类型

该数据类型源自enum8，详见下表183。

表183 StepModeEnum 类型

| 值 | 名称   | 概述   | 一致性 |
|---|------|------|-----|
| 1 | Up   | 向上迈进 | M   |
| 3 | Down | 向下走  | M   |

#### 6.4.3.6.10 ColorLoopActionEnum 类型

该数据类型源自enum8，详见下表184。

表184 ColorLoopActionEnum 类型

| 值 | 名称                                    | 概述                                    | 一致性 |
|---|---------------------------------------|---------------------------------------|-----|
| 0 | Deactivate                            | 取消颜色循环                                | M   |
| 1 | ActivateFromColorLoopStartEnhancedHue | 从ColorLoopStartEnhancedHue字段中的值激活颜色循环 | M   |
| 2 | ActivateFromEnhancedCurrentHue        | 从EnhancedCurrentHue属性的值激活颜色循环         | M   |

#### 6.4.3.6.11 ColorLoopDirectionEnum 类型

该数据类型源自enum8，详见下表185。

表185 ColorLoopDirectionEnum 类型

| 值 | 名称        | 概述         | 一致性 |
|---|-----------|------------|-----|
| 0 | Decrement | 减少颜色循环中的色调 | M   |
| 1 | Increment | 在颜色循环中增加色调 | M   |

#### 6.4.3.7 属性

颜色控制功能集属性见下表186。

表186 颜色控制功能集属性表

| 标识符    | 名称                     | 类型                    | 约束条件     | 质量等级 | 默认               | 通道   | 一致性                    |
|--------|------------------------|-----------------------|----------|------|------------------|------|------------------------|
| 0x0000 | CurrentHue             | uint8                 | max254   | PNQ  | 0                | RV   | HS                     |
| 0x0001 | CurrentSaturation      | uint8                 | max254   | PSNQ | 0                | RV   | HS                     |
| 0x0002 | RemainingTime          | uint16                | max65534 | Q    | 0                | RV   | 0                      |
| 0x0003 | CurrentX               | uint16                | max65279 | PSNQ | 24939<br>(0.381) | RV   | xy                     |
| 0x0004 | CurrentY               | uint16                | max65279 | PSNQ | 24701<br>(0.377) | RV   | xy                     |
| 0x0005 | DriftCompensation      | DriftCompensationEnum | all      | —    | —                | RV   | 0                      |
| 0x0006 | CompensationText       | string                | max254   | —    | —                | RV   | 0                      |
| 0x0007 | ColorTemperatureMireds | uint16                | max65279 | PSNQ | 250<br>(4000K)   | RV   | CT                     |
| 0x0008 | ColorMode              | ColorModeEnum         | all      | N    | 1                | RV   | M                      |
| 0x000F | Options                | OptionsBitmap         | desc     |      | 0                | RWVO | M                      |
| 0x0010 | NumberOfPrimaries      | uint8                 | max6     | FX   | —                | RV   | M                      |
| 0x0011 | Primary1X              | uint16                | max65279 | F    | —                | RV   | NumberOfPrimaries>0, 0 |
| 0x0012 | Primary1Y              | uint16                | max65279 | F    | —                | RV   | NumberOfPrimaries>0, 0 |

表186 颜色控制功能集属性表（续）

| 标识符    | 名称                   | 类型                    | 约束条件     | 质量等级 | 默认 | 通道   | 一致性                    |
|--------|----------------------|-----------------------|----------|------|----|------|------------------------|
| 0x0013 | Primary1Intensity    | uint8                 | all      | FX   | —  | RV   | NumberOfPrimaries>0, 0 |
| 0x0015 | Primary2X            | uint16                | max65279 | F    | —  | RV   | NumberOfPrimaries>1, 0 |
| 0x0016 | Primary2Y            | uint16                | max65279 | F    | —  | RV   | NumberOfPrimaries>1, 0 |
| 0x0017 | Primary2Intensity    | uint8                 | all      | FX   | —  | RV   | NumberOfPrimaries>1, 0 |
| 0x0019 | Primary3X            | uint16                | max65279 | F    | —  | RV   | NumberOfPrimaries>2, 0 |
| 0x001A | Primary3Y            | uint16                | max65279 | F    | —  | RV   | NumberOfPrimaries>2, 0 |
| 0x001B | Primary3Intensity    | uint8                 | all      | FX   | —  | RV   | NumberOfPrimaries>2, 0 |
| 0x0020 | Primary4X            | uint16                | max65279 | F    | —  | RV   | NumberOfPrimaries>3, 0 |
| 0x0021 | Primary4Y            | uint16                | max65279 | F    | —  | RV   | NumberOfPrimaries>3, 0 |
| 0x0022 | Primary4Intensity    | uint8                 | all      | FX   | —  | RV   | NumberOfPrimaries>3, 0 |
| 0x0024 | Primary5X            | uint16                | max65279 | F    | —  | RV   | NumberOfPrimaries>4, 0 |
| 0x0025 | Primary5Y            | uint16                | max65279 | F    | —  | RV   | NumberOfPrimaries>4, 0 |
| 0x0026 | Primary5Intensity    | uint8                 | all      | FX   | —  | RV   | NumberOfPrimaries>4, 0 |
| 0x0028 | Primary6X            | uint16                | max65279 | F    | —  | RV   | NumberOfPrimaries>5, 0 |
| 0x0029 | Primary6Y            | uint16                | max65279 | F    | —  | RV   | NumberOfPrimaries>5, 0 |
| 0x002A | Primary6Intensity    | uint8                 | all      | FX   | —  | RV   | NumberOfPrimaries>5, 0 |
| 0x0030 | WhitePointX          | uint16                | max65279 | —    | —  | RWVM | 0                      |
| 0x0031 | WhitePointY          | uint16                | max65279 | —    | —  | RWVM | 0                      |
| 0x0032 | ColorPointRX         | uint16                | max65279 | —    | —  | RWVM | 0                      |
| 0x0033 | ColorPointRY         | uint16                | max65279 | —    | —  | RWVM | 0                      |
| 0x0034 | ColorPointRIntensity | uint8                 | all      | X    | —  | RWVM | 0                      |
| 0x0036 | ColorPointGX         | uint16                | max65279 | —    | —  | RWVM | 0                      |
| 0x0037 | ColorPointGY         | uint16                | max65279 | —    | —  | RWVM | 0                      |
| 0x0038 | ColorPointGIntensity | uint8                 | all      | X    | —  | RWVM | 0                      |
| 0x003A | ColorPointBX         | uint16                | max65279 | —    | —  | RWVM | 0                      |
| 0x003B | ColorPointBY         | uint16                | max65279 | —    | —  | RWVM | 0                      |
| 0x003C | ColorPointBIntensity | uint8                 | all      | X    | —  | RWVM | 0                      |
| 0x4000 | EnhancedCurrentHue   | uint16                | all      | SNQ  | 0  | RV   | EHUE                   |
| 0x4001 | EnhancedColorMode    | EnhancedColorModeEnum | all      | SN   | 1  | RV   | M                      |
| 0x4002 | ColorLoopActive      | uint8                 | max1     | SN   | 0  | RV   | CL                     |
| 0x4003 | ColorLoopDirection   | uint8                 | max1     | SN   | 0  | RV   | CL                     |

表186 颜色控制功能集属性表（续）

| 标识符    | 名称                              | 类型                      | 约束条件                                               | 质量等级 | 默认   | 通道   | 一致性                       |
|--------|---------------------------------|-------------------------|----------------------------------------------------|------|------|------|---------------------------|
| 0x4004 | ColorLoopTime                   | uint16                  | all                                                | SN   | 25   | RV   | CL                        |
| 0x4005 | ColorLoopStartEnhancedHue       | uint16                  | all                                                | —    | 8960 | RV   | CL                        |
| 0x4006 | ColorLoopStoredEnhancedHue      | uint16                  | all                                                | —    | 0    | RV   | CL                        |
| 0x400A | ColorCapabilities               | ColorCapabilitiesBitmap | max0x001F                                          | —    | 0    | RV   | M                         |
| 0x400B | ColorTempPhysicalMinMireds      | uint16                  | 1to65279                                           | —    | —    | RV   | CT                        |
| 0x400C | ColorTempPhysicalMaxMireds      | uint16                  | max65279                                           | —    | —    | RV   | CT                        |
| 0x400D | CoupleColorTempToLevelMinMireds | uint16                  | ColorTempPhysicalMinMiredstoColorTemperatureMireds | —    | MS   | RV   | CT ColorTemperatureMireds |
| 0x4010 | StartUpColorTemperatureMireds   | uint16                  | 1to65279                                           | NX   | MS   | RWVM | CT ColorTemperatureMireds |

其中，

——关于场景表扩展

如果在同一端点上实现场景管理服务端功能集，则以下属性应成为场景表的ExtensionFieldSetStruct的一部分。如果未实现某个属性，则将在AttributeValuePairStruct中为其使用的值在括号中给出。如果使用ExtensionFieldSetStruct的隐式形式，则AttributeValueList中的属性顺序按给定顺序排列，即首先添加列为1的属性：

1. CurrentX (0)
2. CurrentY (0)
3. EnhancedCurrentHue (空)
4. CurrentSaturation (空)
5. ColorLoopActive (0)
6. ColorLoopDirection (0)
7. ColorLoopTime (0)
8. ColorTemperatureMireds (空)
9. EnhancedColorMode

由于ColorTemperatureMireds和xy之间存在直接关系，因此色温（如果支持）将作为xy存储在场景表中。

场景表中不受设备支持的属性（根据FeatureMap属性）应存在于场景表中但被忽略。

如果场景管理群集服务端在同一端点上实现，并且群集修订版本为6或更高版本，则上表中的所有扩展字段（包括EnhancedColorMode（第9项属性））都应存在于场景的扩展字段集中，场景才被视为有效。这消除了场景中使用的色彩模式的歧义。

——CurrentHue 属性

CurrentHue属性包含光的当前色调值。在更改色调的命令期间，它会尽可能快地更新。以度为单位的色调应与CurrentHue属性有以下关系，如公式（2）所示：

$$Hue = "CurrentHue" * 360/254 \dots\dots\dots (2)$$

式中： CurrentHue 范围是 0 至 254（包括 0 和 254）。

仅当下列情况下，对此属性的更改才应标记为可报告：

每秒最多一次或在运动/过渡结束时。

——CurrentSaturation 属性

此属性应指示光的当前饱和度值。在改变饱和度的命令期间，它会尽可能快地更新。

饱和度（范围从0.0到1.0）应通过以下关系与CurrentSaturation属性相关，如公式（3）所示：

$$\text{Saturation} = \text{"CurrentSaturation"} / 254 \dots\dots\dots (3)$$

式中：CurrentSaturation 的范围是 0 至 254（包括 0 和 254）。

仅当下列情况下，对此属性的更改才应标记为可报告：

- 每秒最多一次或
- 在运动/过渡结束时。

——RemainingTime 属性

此属性应指示当前活动命令导致的转换完成所需的剩余时间（以1/10秒为单位）。仅当下列情况下，对此属性的更改才应标记为可报告：

当它从0变为大于10的任何值时，或者

当它因调用命令而发生变化（增量大于10）时，或者

当它变为0时。

对于转换时间或转换时间变化小于1秒的命令，不得报告此属性的变化。

由于在常规倒计时期间不会报告此属性，因此客户端不应依赖此属性的报告来跟踪剩余持续时间。

——CurrentX 属性

此属性应指示CIExyY颜色空间中定义的标准色度值x的当前值。在更改颜色的命令期间，它会尽可能快地更新。

x的值应与CurrentX属性通过以下关系相关联，如公式（4）所示：

$$X = \text{"CurrentX"} / 65536 \dots\dots\dots (4)$$

式中：CurrentX 的范围是 0 至 65279（含）。

仅当下列情况下，对此属性的更改才应标记为可报告：

- 每秒最多一次或
- 在运动/过渡结束时。

——CurrentY 属性

此属性应指示CIExyY颜色空间中定义的标准色度值y的当前值。在更改颜色的命令期间，它会尽可能快地更新。

y的值应与CurrentY属性通过以下关系相关联，如公式（5）所示：

$$y = \text{"CurrentY"} / 65536 \dots\dots\dots (5)$$

式中：CurrentY 的范围是 0 至 65279（含）。

仅当下列情况下，对此属性的更改才应标记为可报告：

每秒最多一次或在运动/过渡结束时。

——DriftCompensation 属性

此属性应指示使用何种机制（如果有）来补偿随时间的颜色/强度漂移。

——CompensationText 属性

此属性应包含一个文本指示，说明使用哪种机制（如果有）来补偿随时间推移的颜色/强度漂移。

——ColorTemperatureMireds 属性

此属性应指示色温当前值的倒数。ColorTemperatureMireds的单位是mired（微倒数度），又称mirek（微倒数开尔文）。在更改颜色的命令期间，它会尽可能快地更新。

仅当下列情况下，对此属性的更改才应标记为可报告：

- 每秒最多一次或
- 在运动/过渡结束时。

以开尔文为单位的色温值应与mired中的ColorTemperatureMireds属性通过以下关系相关联，如公式（6）所示：

$$\text{Colortemperature}[K] = 1,000,000 / \text{ColorTemperatureMireds} \dots\dots\dots (6)$$

式中：ColorTemperatureMireds 的范围是 1 到 65279（含 1 和 65279），色温范围是 1,000,000K 到 15.32K。

如果实现了该属性，那么也应实现ColorMode属性。

——ColorMode 属性

此属性应指示哪些属性当前正在确定设备的颜色。

ColorMode属性的值不能直接写入-在接收到章节“6.3.5.9 命令”中的任何命令时，它会被设置为该命令的适当模式。

——Options 属性

此属性应指示一个位图，该位图决定了某些群集命令的默认行为。每个依赖于Options属性的命令都应首先构造一个临时Options位图，该位图在命令处理期间有效。临时Options位图具有与Options属性相同的格式和含义，但包括可能被命令字段覆盖的任何位。

此属性仅在调试期间才可以更改。

以下是Options属性和临时Options位图的格式和说明以及对依赖命令的影响。如果所有以下条件都为真，则命令执行将不会继续超出选项处理的范围：

- 开/关功能集与此功能集位于同一端点。
- 此端点上的On/Off群集的OnOff属性为FALSE。
- ExecuteIfOff位的值为0。

——ExecuteIfOff 位

0=如果On/Off功能集中的OnOff属性为FALSE，则不执行命令。

1=如果On/Off功能集中的OnOff属性为FALSE，则执行命令。

——EnhancedCurrentHue 属性

此属性应指示沿CIE1931colortriangle的非等距步骤，并提供16位精度。

此属性的高8位应用作实现特定xy查找表中的索引，以提供非等距步骤。低8位应以线性方式在这些步骤之间进行插值，以便为用户提供颜色缩放。

为了与不支持EHUE的客户端兼容，CurrentHue属性应包含一个根据EnhancedCurrentHue属性计算得出的0到254范围内的色调值。

仅当下列情况下，对此属性的更改才应标记为可报告：

- 每秒最多一次或
- 在运动/过渡结束时。

——EnhancedColorMode 属性

此属性应指示哪些属性当前正在确定设备的颜色。

为了与不支持EHUE的客户端兼容，当灯光使用EnhancedCurrentHue属性时，原始ColorMode属性应指示CurrentHue和CurrentSaturation。如果ColorMode属性发生更改，则其新值应复制到EnhancedColorMode属性。

——ColorLoopActive 属性

此属性应指示颜色循环的当前活动状态。如果此属性的值为0，则颜色循环不应处于活动状态。如果此属性的值为1，则颜色循环应处于活动状态。

——ColorLoopDirection 属性

此属性应指示颜色循环的当前方向。如果此属性的值为0，则EnhancedCurrentHue属性应减少。如果此属性的值为1，则EnhancedCurrentHue属性应增加。

——ColorLoopTime 属性

此属性应指示执行全彩循环所需的秒数，即循环执行EnhancedCurrentHue属性的所有值（0至65534之间）。

——ColorLoopStartEnhancedHue 属性

此属性应指示应从其开始颜色循环的EnhancedCurrentHue属性的值。

——ColorLoopStoredEnhancedHue 属性

此属性应指示颜色循环开始之前EnhancedCurrentHue属性的值。颜色循环完成后，EnhancedCurrentHue属性应恢复为此值。

——ColorCapabilities 属性

此属性应指示设备的颜色控制能力。

ColorCapabilities属性的0-4位应具有与FeatureMap属性的相应位相同的值。ColorCapabilities中的所有其他位应为0。

——ColorTempPhysicalMinMireds 属性

此属性应指示硬件支持的最小mired值。ColorTempPhysicalMinMireds对应于硬件支持的最大色温（以开尔文为单位）。

ColorTempPhysicalMinMireds≤ColorTemperatureMireds。

——ColorTempPhysicalMaxMireds 属性

此属性应指示硬件支持的最大mired值。ColorTempPhysicalMaxMireds对应于硬件支持的最低色温（以开尔文为单位）。

ColorTemperatureMireds≤ColorTempPhysicalMaxMireds

——CoupleColorTempToLevelMinMireds 属性

当级别控制群集的选项属性的CoupleColorTempToLevel位等于1时，此属性应指示ColorTemperatureMireds属性值的下限，以便将ColorTemperatureMireds属性与CurrentLevel属性耦合。当将ColorTemperatureMireds属性与CurrentLevel属性耦合时，该值应对应于CurrentLevel值254（100%）。

应设置此属性，以便存在以下关系：

ColorTempPhysicalMinMireds≤CoupleColorTempToLevelMinMireds≤ColorTemperatureMireds

注：由于此属性存储为微倒数（mired）值（即开尔文色温=1,000,000/CoupleColorTempToLevelMinMireds），CoupleColorTempToLevelMinMireds属性对应于色温值的上限以设备支持的开尔文为单位。

——StartupColorTemperatureMireds 属性

此属性应指示灯通电时应使用的所需启动色温值，并且该值应反映在ColorTemperatureMireds属性中。此外，ColorMode和EnhancedColorMode属性应设置为2（ColorTemperatureMireds）。StartupColorTemperatureMireds属性的值列于下表187中。

表187 StartupColorTemperatureMireds 属性表

| 值        | 开机后的动作                            |
|----------|-----------------------------------|
| 1to65279 | 将ColorTemperatureMireds属性设置为此值    |
| null     | 将ColorTemperatureMireds属性设置为其先前的值 |

——NumberOfPrimaries 属性

此属性应指示此设备上实现的原色数量。值为null应指示原色数量未知。

当实现此属性时，还应为从1到NumberOfPrimaries的每个原色实现以下用于指示原色“x”和“y”颜色值的属性，而不留空隙。Primary1Intensity属性和后续强度属性的实现是可选的。

——Primary1X 属性

此属性应指示此原色的归一化色度值x，如CIExyY颜色空间中所定义。

x的值应与Primary1X属性通过以下关系相关联，如公式（7）所示：

$$x = \text{Primary1X} / 65536 \dots\dots\dots (7)$$

式中：Primary1X 在 0 至 65279 范围内（含）

——Primary1Y 属性

此属性应指示此原色的归一化色度值y，如CIExyY颜色空间中所定义。

y的值应与Primary1Y属性通过以下关系关联，如公式（8）所示：

$$y = \text{Primary1Y} / 65536 \dots\dots\dots (8)$$

式中：Primary1Y 在 0 至 65279 范围内（含）

——Primary1Intensity 属性

此属性应指示该原色的最大强度的表示，如镇流器配置簇（参见章节“6.4.4 镇流器配置功能集”）中的调光光曲线所定义，并进行规范化，使得具有最高最大强度的原色包含值254。

空值应指示该主要值不可用。

——Primary2X、Primary2Y、Primary2Intensity、Primary3X、Primary3Y、

Primary3Intensity、Primary4X、Primary4Y、Primary4Intensity、Primary5X、

Primary5Y、Primary5Intensity、Primary6X、Primary6Y 和 Primary6Intensity 属性

这些属性应代表2nd、3rd、4th、5th和6th原色的能力（如果存在），其方式与Primary1X、Primary1Y和Primary1Intensity属性相同。

——WhitePointX 属性

此属性应指示设备当前白点的标准化色度值x（如CIExyY颜色空间中所定义）。

x的值应与WhitePointX属性通过以下关系相关联： $x = \text{WhitePointX} / 65536$ （WhitePointX在0至65279范围内（含））

——WhitePointY 属性

此属性应指示设备当前白点的归一化色度值y（如CIExyY颜色空间中定义）。

y的值应与WhitePointY属性相关，关系为： $y = \text{WhitePointY} / 65536$ （WhitePointY的范围为0到65279，含）

——ColorPointRX 属性

此属性应指示设备红色点的归一化色度值x（如CIExyY颜色空间中定义）。

x的值应与ColorPointRX属性相关，关系为： $x = \text{ColorPointRX} / 65536$ （ColorPointRX范围为0至65279，含）

——ColorPointRY 属性

此属性应指示设备红色点的归一化色度值y，如CIExyY颜色空间中定义的那样。

y的值应与ColorPointRY属性相关，关系为： $y = \text{ColorPointRY} / 65536$ （ColorPointRY的范围为0到65279，含）。

——ColorPointRIntensity 属性

此属性应指示镇流器配置簇（参见章节“6.4.4 镇流器配置功能集”）中调光曲线定义的红色点的相对强度的表示，并进行归一化，使得具有最高相对强度的色点包含值254。

空值应表示无效值。

——ColorPointGX、ColorPointGY、ColorPointGIntensity、ColorPointBX、ColorPointBY 和 ColorPointBIntensity 属性

这些属性应以与ColorPointRX、ColorPointRY和ColorPointRIntensity属性相同的方式表示绿色和蓝色点的色度值和强度。如果实现了红色、绿色或蓝色点属性中的任何一个，那么它们都应被实现。

#### 6.4.3.8 命令

##### 6.4.3.8.1 命令列表

颜色控制功能集命令见下表188。

表188 颜色控制功能集命令表

| 标识符  | 名称                             | 方向            | 返回 | 通道 | 一致性      |
|------|--------------------------------|---------------|----|----|----------|
| 0x00 | MoveToHue                      | client→server | Y  | 0  | HS       |
| 0x01 | MoveHue                        | client→server | Y  | 0  | HS       |
| 0x02 | StepHue                        | client→server | Y  | 0  | HS       |
| 0x03 | MoveToSaturation               | client→server | Y  | 0  | HS       |
| 0x04 | MoveSaturation                 | client→server | Y  | 0  | HS       |
| 0x05 | StepSaturation                 | client→server | Y  | 0  | HS       |
| 0x06 | MoveToHueAndSaturation         | client→server | Y  | 0  | HS       |
| 0x07 | MoveToColor                    | client→server | Y  | 0  | xy       |
| 0x08 | MoveColor                      | client→server | Y  | 0  | xy       |
| 0x09 | StepColor                      | client→server | Y  | 0  | xy       |
| 0x0A | MoveToColorTemperature         | client→server | Y  | 0  | CT       |
| 0x40 | EnhancedMoveToHue              | client→server | Y  | 0  | EHUE     |
| 0x41 | EnhancedMoveHue                | client→server | Y  | 0  | EHUE     |
| 0x42 | EnhancedStepHue                | client→server | Y  | 0  | EHUE     |
| 0x43 | EnhancedMoveToHueAndSaturation | client→server | Y  | 0  | EHUE     |
| 0x44 | ColorLoopSet                   | client→server | Y  | 0  | CL       |
| 0x47 | StopMoveStep                   | client→server | Y  | 0  | HS xy CT |
| 0x4B | MoveColorTemperature           | client→server | Y  | 0  | CT       |
| 0x4C | StepColorTemperature           | client→server | Y  | 0  | CT       |

##### 6.4.3.8.2 通用使用说明

当通过这些命令之一要求更改颜色时，实现应在设备硬件的限制范围内选择一种尽可能接近请求的颜色。确定颜色的真实表示超出了本文件的范围。但是，只要收到的命令的颜色数据字段在本文件允许的范围内，并且没有错误条件适用，则生成的状态代码应为成功。

例如MoveToColorTemperature命令：如果硬件无法实现目标色温，则当设备达到该色温时（可能在请求的转换时间之前），色温应当被限制在可实现的物理最小值或最大值（取决于色温转换的方向）。

如果颜色循环处于活动状态（即ColorLoopActive属性等于1），则只能通过发送特定的ColorLoopSet命令帧并请求停用颜色循环来停止颜色循环（即，在收到具有“停止”语义的另一个命令（例如带有MoveMode==Stop的EnhancedMoveToHue命令或StopMoveStep命令）时，不得停止颜色循环）。此外，当颜色循环处于活动状态时，制造商可以选择忽略影响色调变化的传入颜色命令。

#### 6.4.3.8.3 关于 ColorMode 更改的注意事项

收到表\*\*中的任一命令时，首先要将ColorMode属性更改为适合该命令的值（请参阅“6.5.3.9.1 命令”）。

请注意，当从一种颜色模式切换到另一种颜色模式时（例如，从CurrentX/CurrentY切换到CurrentHue/CurrentSaturation），命令的起始颜色是通过计算新属性的值（在本例中为CurrentHue、CurrentSaturation）和旧属性的值（在本例中为CurrentX和CurrentY）形成的。

当从一种模式切换到另一种颜色范围更受限的模式时（例如，从CurrentX/CurrentY切换到CurrentHue/CurrentSaturation，或从CurrentHue/CurrentSaturation切换到ColorTemperatureMireds），当前颜色值可能在新模式下没有等效值。此类情况下的行为取决于制造商，因此建议避免此类颜色模式在使用过程中会发生改变。

#### 6.4.3.8.4 OptionsMask 和 OptionsOverride 字段的使用

OptionsMask和OptionsOverride字段均应存在。提供默认值来解释旧设备中缺失的字段。应使用OptionsMask和OptionsOverride字段从Options属性创建临时Options位图。临时Options位图的每一位应按如下方式确定：

Options属性中的每一位应确定临时Options位图中的相应位，除非OptionsMask字段存在且相应位设置为1，在这种情况下OptionsOverride字段中的相应位应确定临时Options位图中的相应位。

然后应按照Options的定义处理生成的临时选项位图。

#### 6.4.3.8.5 MoveToHue 命令

MoveToHue 命令信息详见下表189。

表189 MoveToHue 命令表

| 标识符 | 名称              | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|----------|------|----|-----|
| 0   | Hue             | uint8         | max254   | —    | —  | M   |
| 1   | Direction       | DirectionEnum | all      | —    | —  | M   |
| 2   | TransitionTime  | uint16        | max65534 | —    | —  | M   |
| 3   | OptionsMask     | OptionsBitmap | desc     | —    | 0  | M   |
| 4   | OptionsOverride | OptionsBitmap | desc     | —    | 0  | M   |

其中，

- HueField字段应指示要移动的色调。
- DirectionField字段应指示移动方向。
- TransitionTime字段应指示以十分之一秒为单位指示移动到新色调所需的时间。
- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

在收到此命令时，设备还应将ColorMode属性设置为值0，然后应从其当前色调移动到色调字段中给出的值。

移动应是连续的，即不是阶跃函数，并且移动到新色调所花费的时间应等于TransitionTime字段。

由于色调实际上是在圆周上测量的，因此新的色调可以向任一方向移动。色调变化的方向由方向字段给出。如果方向是“最短距离”，则采用涉及圆周最短路径的方向。这种情况对应于预期的正常使用。如果方向是“最长距离”，则采用涉及圆周最长路径的方向。这种情况可用于“彩虹效果”。在这两种情况下，如果两个距离相同，则应采用向上方向。

#### 6.4.3.8.6 MoveHue 命令

MoveHue 命令信息详见下表190。

表190 MoveHue 命令表

| 标识符 | 名称              | 类型            | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|------|------|----|-----|
| 0   | MoveMode        | MoveModeEnum  | all  | –    | –  | M   |
| 1   | Rate            | uint8         | all  | –    | –  | M   |
| 2   | OptionsMask     | OptionsBitmap | desc | –    | 0  | M   |
| 3   | OptionsOverride | OptionsBitmap | desc | –    | 0  | M   |

其中，

- MoveMode字段应指示移动模式，详见下表191。
- Rate字段应以每秒步数为单位指示移动速率。一步是设备色调的一个单位变化。
- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，

如果MoveMode字段设置为1（向上）或3（向下）：

- 如果Rate字段的值为零，则该命令无效，并且应返回响应，并将状态代码设置为INVALID\_COMMAND。
- 否则，设备应将ColorMode属性设置为值0（CurrentHueAndCurrentSaturation），然后应从其当前色调以连续的方式向上或向下移动，如表中所述。

如果MoveMode字段设置为0（停止），则Rate字段应被完全忽略。

表191 MoveMode 字段

| MoveMode | 收到命令后的动作                                            |
|----------|-----------------------------------------------------|
| Stop     | 如果移动，则停止，否则忽略命令（即命令被接受但没有效果）。这应停止任何正在进行的色调和/或饱和度转换  |
| Up       | 按照速率字段中给出的速率增加设备的色调。如果色调达到设备允许的最大值，则绕回并从其允许的最小值继续进行 |
| Down     | 按照“速率”字段中给出的速率降低设备的色调。如果色调达到设备允许的最小值，则绕回并从其允许的最大值继续 |

#### 6.4.3.8.7 StepHue 命令

StepHue 命令信息详见下表192。

表192 StepHue 命令表

| 标识符 | 名称              | 类型            | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|------|------|----|-----|
| 0   | StepMode        | StepModeEnum  | all  | –    | –  | M   |
| 1   | StepSize        | uint8         | all  | –    | –  | M   |
| 2   | TransitionTime  | uint8         | all  | –    | –  | M   |
| 3   | OptionsMask     | OptionsBitmap | desc | –    | 0  | M   |
| 4   | OptionsOverride | OptionsBitmap | desc | –    | 0  | M   |

其中，

- StepMode字段应指示要执行的步骤的模式已确认。
- StepSize字段应指示要添加到（或从）设备当前色调值中减去的变化。

- TransitionTime字段应以十分之一秒为单位指示执行该步骤所需的时间。一步是设备色调以步长为单位的变化。此处，TransitionTime数据字段的数据类型为uint8，而其他功能集/命令中的TransitionTime数据字段更常使用uint16。
- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，如果StepSize字段的值为零，则该命令无效，并且应返回响应，并将状态代码设置为INVALID\_COMMAND。

否则，设备应将ColorMode属性设置为值0（CurrentHueAndCurrentSaturation），然后应从其当前色调向上或向下移动一步，如表193中所述。

表193 StepMode 字段

| StepMode | 收到命令后的动作                                       |
|----------|------------------------------------------------|
| Up       | 以连续的方式逐级增加设备的色调。<br>如果色调值达到最大值，则回溯并从允许的最小值继续进行 |
| Down     | 以连续的方式逐级降低设备的色调。如果色调值达到最小值，则回溯并从允许的最大值继续       |

#### 6.4.3.8.8 MoveToSaturation 命令

MoveToSaturation 命令信息详见下表194。

表194 MoveToSaturation 命令表

| 标识符 | 名称              | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|----------|------|----|-----|
| 0   | Saturation      | uint8         | max254   | —    | —  | M   |
| 1   | TransitionTime  | uint16        | max65534 | —    | —  | M   |
| 2   | OptionsMask     | OptionsBitmap | desc     | —    | 0  | M   |
| 3   | OptionsOverride | OptionsBitmap | desc     | —    | 0  | M   |

其中，OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，设备应将ColorMode属性设置为值0（CurrentHueAndCurrentSaturation），然后应从其当前饱和度移动到饱和度字段中给出的值。

移动应是连续的，即不是阶跃函数，并且移动到新的饱和度所需的时间应等于TransitionTime字段，以1/10秒为单位。

#### 6.4.3.8.9 MoveSaturation 命令

MoveSaturation 命令信息详见下表195。

表195 MoveSaturation 命令表

| 标识符 | 名称              | 类型            | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|------|------|----|-----|
| 0   | MoveMode        | MoveModeEnum  | all  | —    | —  | M   |
| 1   | Rate            | uint8         | all  | —    | —  | M   |
| 2   | OptionsMask     | OptionsBitmap | desc | —    | 0  | M   |
| 3   | OptionsOverride | OptionsBitmap | desc | —    | 0  | M   |

其中，

- MoveMode字段应指示移动模式，详见下表196，如MoveHue命令中所述。
- Rate字段应以每秒步数为单位指示移动速率。一步是设备饱和度的一个单位变化。
- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，

如果MoveMode字段设置为1（向上）或3（向下）

- 如果Rate字段的值为零，则命令无效，并且应返回响应，其中状态代码设置为INVALID\_COMMAND。

– 否则，设备应将ColorMode属性设置为值0（CurrentHueAndCurrentSaturation），然后应以连续的方式从其当前饱和度向上或向下移动，如表中所述。

如果MoveMode字段设置为0（停止），则Rate字段应被完全忽略。

表196 MoveMode 字段表

| MoveMode | 收到命令后的动作                                          |
|----------|---------------------------------------------------|
| Stop     | 如果移动，则停止，否则忽略命令（即命令被接受但没有效果，这应停止任何正在进行的色调和/或饱和度转换 |
| Up       | 按照Rate字段中给出的速率增加设备的饱和度，如果饱和度达到设备允许的最大值，则停止        |
| Down     | 按照Rate字段中给出的速率降低设备的饱和度，如果饱和度达到设备允许的最小值，则停止        |

#### 6.4.3.8.10 StepSaturation 命令

StepSaturation 命令信息详见下表197。

表197 StepSaturation 命令表

| 标识符 | 名称              | 类型            | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|------|------|----|-----|
| 0   | StepMode        | StepModeEnum  | all  | –    | –  | M   |
| 1   | StepSize        | uint8         | all  | –    | –  | M   |
| 2   | TransitionTime  | uint8         | all  | –    | –  | M   |
| 3   | OptionsMask     | OptionsBitmap | desc | –    | 0  | M   |
| 4   | OptionsOverride | OptionsBitmap | desc | –    | 0  | M   |

其中，

- StepMode字段应指示要执行的步骤的模式，如StepHue命令中所述。
- StepSize字段应指示要添加到设备饱和度当前值（或从中减去）的变化。
- TransitionTime字段应以十分之一秒为单位，指示执行该步骤所需的时间。步骤是设备步长单位饱和度的变化。

注：此处，TransitionTime数据字段的数据类型为uint8，而其他功能集/命令中的TransitionTime数据字段更常使用uint16。

- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，如果StepSize字段的值为零，则该命令无效，并且应返回响应，并将状态代码设置为INVALID\_COMMAND。

否则，设备应将ColorMode属性设置为值0（CurrentHueAndCurrentSaturation），然后应从其当前饱和度向上或向下移动一步。详见下表198。

表198 StepMode 字段

| StepMode | 收到命令后的动作                               |
|----------|----------------------------------------|
| Up       | 持续将设备的饱和度提高一步。但是，如果饱和度值已经是最大值，则不执行任何操作 |
| Down     | 持续将设备的饱和度降低一步。但是，如果饱和度值已经是最小值，则不执行任何操作 |

#### 6.4.3.8.11 MoveToHueAndSaturation 命令

MoveToHueAndSaturation 命令信息详见下表199。

表199 MoveToHueAndSaturation 命令表

| 标识符 | 名称              | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|----------|------|----|-----|
| 0   | Hue             | uint8         | max254   | —    | —  | M   |
| 1   | Saturation      | uint8         | max254   | —    | —  | M   |
| 2   | TransitionTime  | uint16        | max65534 | —    | —  | M   |
| 3   | OptionsMask     | OptionsBitmap | desc     | —    | 0  | M   |
| 4   | OptionsOverride | OptionsBitmap | desc     | —    | 0  | M   |

其中,OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后,设备应将ColorMode属性设置为值0 (CurrentHueAndCurrentSaturation), 然后应从其当前色调和饱和度移动到色调和饱和度字段中给出的值。

移动应是连续的, 即不是阶跃函数, 并且移动到新颜色所花费的时间应等于TransitionTime字段, 以1/10秒为单位。

未指定转换期间所采用的通过颜色空间的路径, 但建议采用通过颜色空间的最短路径, 即, 移动是穿过CIExyY颜色空间的“直线”。

6.4.3.8.12 MoveToColor 命令

MoveToColor 命令信息详见下表200。

表200 MoveToColor 命令表

| 标识符 | 名称              | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|----------|------|----|-----|
| 0   | ColorX          | uint16        | max65279 | —    | —  | M   |
| 1   | ColorY          | uint16        | max65279 | —    | —  | M   |
| 2   | TransitionTime  | uint16        | max65534 | —    | —  | M   |
| 3   | OptionsMask     | OptionsBitmap | desc     | —    | 0  | M   |
| 4   | OptionsOverride | OptionsBitmap | desc     | —    | 0  | M   |

其中,OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

在收到此命令时,设备应将ColorMode属性的值(如果已实现)设置为1 (CurrentXAndCurrentY), 然后应从其当前颜色移动到ColorX和ColorY字段中给定的颜色。

移动应是连续的, 即不是阶跃函数, 并且移动到新颜色所花费的时间应等于TransitionTime字段, 以1/10秒为单位。

未指定转换期间所采用的通过颜色空间的路径, 但建议采用通过颜色空间的最短路径, 即, 移动是穿过CIExyY颜色空间的“直线”。

6.4.3.8.13 MoveColor 命令

IMoveColor 命令信息详见下表201。

表201 IMoveColor 命令表

| 标识符 | 名称              | 类型            | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|------|------|----|-----|
| 0   | RateX           | int16         | all  | —    | —  | M   |
| 1   | RateY           | int16         | all  | —    | —  | M   |
| 2   | OptionsMask     | OptionsBitmap | desc | —    | 0  | M   |
| 3   | OptionsOverride | OptionsBitmap | desc | —    | 0  | M   |

其中,

- RateX字段应指示移动速率(以步数/秒为单位)。一步是设备CurrentX属性中一个单位的变化。

- RateY字段应指示移动速率（以步数/秒为单位）。一步是设备CurrentY属性的一个单位变化。
- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，如果 RateX 和 RateY 字段均为零，则不应进行任何移动，并且命令执行不应产生任何效果，除了停止此集群之前接收到的任何命令的操作。虽然此命令（当RateX和RateY字段都设置为零时）可用于停止此功能集的任何操作，但建议改用StopMoveStep命令来实现此目标。否则，设备应将ColorMode属性的值（如果已实现）设置为1（CurrentX And CurrentY），然后应根据指定的速率以连续方式从其当前颜色移动。此移动应持续到此设备无法实现下一步的目标颜色为止。

6.4.3.8.14 StepColor 命令

StepColor 命令信息详见下表202。

表202 StepColor 命令表

| 标识符 | 名称              | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|----------|------|----|-----|
| 0   | StepX           | int16         | all      | –    | –  | M   |
| 1   | StepY           | int16         | all      | –    | –  | M   |
| 2   | TransitionTime  | uint16        | max65534 | –    | –  | M   |
| 3   | OptionsMask     | OptionsBitmap | desc     | –    | 0  | M   |
| 4   | OptionsOverride | OptionsBitmap | desc     | –    | 0  | M   |

其中，

- StepX和StepY字段应分别指示要添加到设备的CurrentX属性和CurrentY属性的更改。
- TransitionTime字段应以十分之一秒为单位指示执行颜色变化所需的时间。
- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，如果StepX和StepY字段都包含零值，则该命令无效，并且应返回响应，并将状态代码设置为INVALID\_COMMAND。

否则，设备应将ColorMode属性的值（如果实现）设置为1（CurrentX And CurrentY），然后应从其当前颜色按照指示的颜色步骤移动。

移动应是连续的，即不是阶跃函数，并且移动到新颜色所花费的时间应等于TransitionTime字段，以1/10秒为单位。

未指定转换期间所采用的通过颜色空间的路径，但建议采用通过颜色空间的最短路径，即，移动是穿过CIExyY颜色空间的“直线”。

注：如果所需步长大于有符号的16位整数所能表示的步长，则应该发出多个步长命令。

6.4.3.8.15 MoveToColorTemperature 命令

MoveToColor Temperature 命令信息详见下表203。

表203 MoveToColor Temperature 命令表

| 标识符 | 名称                     | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|------------------------|---------------|----------|------|----|-----|
| 0   | ColorTemperatureMireds | uint16        | max65279 | –    | –  | M   |
| 1   | TransitionTime         | uint16        | max65534 | –    | –  | M   |
| 2   | OptionsMask            | OptionsBitmap | desc     | –    | 0  | M   |
| 3   | OptionsOverride        | OptionsBitmap | desc     | –    | 0  | M   |

其中，OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

在收到此命令时，设备应将ColorMode属性的值（如果已实现）设置为2（ColorTemperatureMireds），然后应从其当前颜色移动到ColorTemperatureMired字段给出的颜色。

移动应是连续的，即不是阶跃函数，并且移动到新颜色所花费的时间应等于TransitionTime字段，以1/10秒为单位。

根据此颜色模式的定义，转换过程中穿过颜色空间的路径沿着“黑体线（Black Body Line）”。

如果ColorTemperatureMireds字段小于ColorTempPhysicalMinMireds属性的值，则ColorTempPhysicalMinMireds属性的值应用作ColorTemperatureMireds属性的最终目标。

如果ColorTemperatureMireds字段大于ColorTempPhysicalMaxMireds属性的值，则ColorTempPhysicalMaxMireds属性的值应用作ColorTemperatureMireds属性的最终目标。

#### 6.4.3.8.16 EnhancedMoveToHue 命令

EnhancedMoveToHue命令允许光线从当前色调平滑连续地过渡到目标色调，详见下表204。

表204 EnhancedMoveToHue 命令表

| 标识符 | 名称              | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|----------|------|----|-----|
| 0   | EnhancedHue     | uint16        | all      | —    | —  | M   |
| 1   | Direction       | DirectionEnum | all      | —    | —  | M   |
| 2   | TransitionTime  | uint16        | max65534 | —    | —  | M   |
| 3   | OptionsMask     | OptionsBitmap | desc     | —    | 0  | M   |
| 4   | OptionsOverride | OptionsBitmap | desc     | —    | 0  | M   |

其中，

- EnhancedHue字段应指示光的目标扩展色调。
- Direction字段应指示移动方向。
- TransitionTime字段应指示转换时间，如MoveToHue命令中所述。
- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，设备应将ColorMode属性设置为0（CurrentHueAndCurrentSaturation），并将EnhancedColorMode属性设置为值3（EnhancedCurrentHueAndCurrentSaturation）。然后，设备应从其当前增强色调移至EnhancedHue字段中给出的值。

移动应是连续的，即不是阶跃函数，并且移动到新的增强色调所花费的时间应等于TransitionTime字段。

#### 6.4.3.8.17 EnhancedMoveHue 命令

EnhancedMoveHue命令允许灯光从当前色调开始连续过渡，详见下表205。

表205 EnhancedMoveHue 命令表

| 标识符 | 名称              | 类型            | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|------|------|----|-----|
| 0   | MoveMode        | MoveModeEnum  | desc | —    | —  | M   |
| 1   | Rate            | uint16        | all  | —    | —  | M   |
| 2   | OptionsMask     | OptionsBitmap | desc | —    | 0  | M   |
| 3   | OptionsOverride | OptionsBitmap | desc | —    | 0  | M   |

其中，

- MoveMode字段应指示移动模式，如MoveHue命令中所述。
- Rate字段应以每秒步数为单位指示移动速率。一步是设备扩展色调的一个单位变化。
- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，

如果MoveMode字段设置为1（向上）或3（向下）

— 如果Rate字段的值为零，则该命令无效，并且应返回响应，并将状态代码设置为INVALID\_COMMAND。

– 否则，设备应将ColorMode属性设置为0（CurrentHueAndCurrentSaturation），并将EnhancedColorMode属性设置为值3（EnhancedCurrentHueAndCurrentSaturation）。然后，设备应以连续的方式从其当前增强色调向上或向下移动，如表中206所述。

如果MoveMode字段设置为0（停止），则Rate字段应被完全忽略。

表206 MoveMode 字段表

| MoveMode | 收到命令后的动作                                                |
|----------|---------------------------------------------------------|
| Stop     | 如果移动，则停止，否则忽略命令（即命令被接受但没有效果）。这应停止任何正在进行的色调和/或饱和度转换      |
| Up       | 按照速率字段中给出的速率增加设备的增强色调。如果增强色调达到设备允许的最大值，则绕回并从其允许的最小值继续进行 |
| Down     | 按照“速率”字段中给出的速率降低设备的增强色调。如果色调达到设备允许的最小值，则绕回并从其允许的最大值继续进行 |

#### 6.4.3.8.18 EnhancedStepHue 命令

EnhancedStepHue 命令允许光线从其当前色调以分阶过渡的方式移动，从而产生通过XY空间的线性过渡，详见下表207。

表207 EnhancedStepHue 命令表

| 标识符 | 名称              | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|----------|------|----|-----|
| 0   | StepMode        | StepModeEnum  | desc     | –    | –  | M   |
| 1   | StepSize        | uint16        | all      | –    | –  | M   |
| 2   | TransitionTime  | uint16        | max65534 | –    | –  | M   |
| 3   | OptionsMask     | OptionsBitmap | desc     | –    | 0  | M   |
| 4   | OptionsOverride | OptionsBitmap | desc     | –    | 0  | M   |

其中，

- StepMode字段应指示要执行的步骤的模式，如StepHue命令中所述。
- StepSize字段应指示要添加到设备增强色调的当前值（或从中减去）的变化。
- TransitionTime字段应以1/10秒为单位，指示执行该步骤所需的时间。步骤是设备增强色调的改变，其幅度与步长字段相对应。

注：此处 TransitionTime数据字段的数据类型为uint16，而 StepHue命令的 TransitionTime数据字段的数据类型为uint8。

- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，如果StepSize字段的值为零，则该命令无效，并且应返回响应，并将状态代码设置为INVALID\_COMMAND。

否则，设备应将ColorMode属性设置为0（CurrentHueAndCurrentSaturation），并将EnhancedColorMode属性设置为值3（EnhancedCurrentHueAndCurrentSaturation）。然后，设备应从其当前增强色调向上或向下移动一步，如表208所述。

表208 StepMode 字段表

| StepMode | 收到命令后的动作                                       |
|----------|------------------------------------------------|
| Up       | 将设备的增强色调增加一级。如果增强色调达到设备允许的最大值，则绕回并从其允许的最小值继续进行 |
| Down     | 将设备的增强色调降低一级。如果色调达到设备允许的最小值，则绕回并从其允许的最大值继续     |

#### 6.4.3.8.19 EnhancedMoveToHueAndSaturation 命令

EnhancedMoveToHueAndSaturation 命令允许光线从当前色调平滑连续地过渡到目标色调，并从当前饱和度平滑连续地过渡到目标饱和度，详见下表209。

表209 EnhancedMoveToHueAndSaturation 命令表

| 标识符 | 名称              | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|----------|------|----|-----|
| 0   | EnhancedHue     | uint16        | all      | —    | —  | M   |
| 1   | Saturation      | uint8         | max254   | —    | —  | M   |
| 2   | TransitionTime  | uint16        | max65534 | —    | —  | M   |
| 3   | OptionsMask     | OptionsBitmap | desc     | —    | 0  | M   |
| 4   | OptionsOverride | OptionsBitmap | desc     | —    | 0  | M   |

其中，

- EnhancedHue字段应指示光的目标扩展色调。
- Saturation字段应指示饱和度，如MoveToHueAndSaturation命令中所述。
- TransitionTime字段应指示转换时间，如MoveToHue命令中所述。
- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，设备应将ColorMode属性设置为值0（CurrentHueAndCurrentSaturation），并将EnhancedColorMode属性设置为值3（EnhancedCurrentHueAndCurrentSaturation）。然后，设备应从其当前增强色调和饱和度移至EnhancedHue和Saturation字段中给出的值。

移动应是连续的，即不是阶跃函数，移动到新颜色应等于TransitionTime字段，以十分之一秒为单位。

未指定转换期间所采用的通过颜色空间的路径，，但建议通过颜色空间采取最短路径，即在CIE xyY颜色空间中以‘直线’移动。

#### 6.4.3.8.20 ColorLoopSet 命令

ColorLoopSet 命令允许激活色彩循环，使彩色灯光在其色调范围内循环变化，详见下表210。

表210 ColorLoopSet 命令表

| 标识符 | 名称              | 类型                     | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|------------------------|------|------|----|-----|
| 0   | UpdateFlags     | UpdateFlagsBitmap      | all  | —    | —  | M   |
| 1   | Action          | ColorLoopActionEnum    | all  | —    | —  | M   |
| 2   | Direction       | ColorLoopDirectionEnum | all  | —    | —  | M   |
| 3   | Time            | uint16                 | all  | —    | —  | M   |
| 4   | StartHue        | uint16                 | all  | —    | —  | M   |
| 5   | OptionsMask     | OptionsBitmap          | desc | —    | 0  | M   |
| 6   | OptionsOverride | OptionsBitmap          | desc | —    | 0  | M   |

其中，

- UpdateFlags字段应指示在颜色循环启动之前要更新哪些颜色循环属性（从其他字段提供的值，请参阅下面的字段描述）。
- Action字段应指示针对颜色循环采取的操作。
- Direction字段应指示颜色循环的方向。
- Time字段应指示执行全色循环的秒数。
- StartHueField字段应指示颜色循环使用的起始色调。
- OptionsMask和OptionsOverride字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

收到此命令后，设备应首先根据UpdateFlags字段的值更新其颜色循环属性，如下所示。

- 如果 UpdateDirection 子字段设置为 1，则设备应设置 ColorLoopDirection 属性为 Direction 字段的值。
- 如果 UpdateTime 子字段设置为 1，则设备应将 ColorLoopTime 属性设置为 Time 字段的值。
- 如果 UpdateStartHue 子字段设置为 1，则设备应将 ColorLoopStartEnhancedHue 属性设置为 StartHue 字段的值。

- d) 如果 UpdateFlags 字段的 UpdateAction 子字段设置为 1, 则设备应遵守 Action 字段中指定的操作, 如下所示。
- 1) 如果 Action 字段的值设置为 0 且颜色循环处于活动状态, 即 ColorLoopActive 属性设置为 1, 则设备应停用颜色循环, 将 ColorLoopActive 属性设置为 0, 并将 EnhancedCurrentHue 属性设置为 ColorLoopStoredEnhancedHue 属性的值。
  - 2) 如果 Action 字段的值设置为 0, 且颜色循环处于非活动状态, 即 ColorLoopActive 属性设置为 0, 则设备应忽略命令的操作更新组件, 即 EnhancedCurrentHue 属性不得更新 (与上一条中描述的情况形成对比)。
  - 3) 如果 Action 字段的值设置为 1, 则设备应将 ColorLoopStoredEnhancedHue 属性设置为 EnhancedCurrentHue 属性的值, 将 ColorLoopActive 属性设置为 1, 并激活颜色循环, 从 ColorLoopStartEnhancedHue 属性的值开始。
  - 4) 如果 Action 字段的值设置为 2, 则设备应将 ColorLoopStoredEnhancedHue 属性设置为 EnhancedCurrentHue 属性的值, 将 ColorLoopActive 属性设置为 1, 并从 EnhancedCurrentHue 属性的值开始激活颜色循环。

如果颜色循环处于活动状态, 设备应在 ColorLoopTime 属性指定的时间内, 按照 ColorLoopDirection 属性的方向循环 EnhancedCurrentHue 属性的整个值范围。增量/减量级别取决于具体应用。

如果颜色循环处于活动状态 (并保持活动状态), 则设备应立即对 ColorLoopDirection 和 ColorLoopTime 属性的更新做出反应。

#### 6.4.3.8.21 StopMoveStep 命令

提供 StopMoveStep 命令是为了允许停止 MoveTo 和 Step 命令。详见下表 211。

注: 这会自动为 LevelControl 群集提供对称性。

注: StopMoveStep 命令对活动颜色循环没有影响。

表 211 StopMoveStep 命令表

| 标识符 | 名称              | 类型            | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|-----------------|---------------|------|------|----|-----|
| 0   | OptionsMask     | OptionsBitmap | desc | —    | 0  | M   |
| 1   | OptionsOverride | OptionsBitmap | desc | —    | 0  | M   |

其中, OptionsMask 和 OptionsOverride 字段应根据 OptionsMask 和 OptionsOverride 字段的使用进行处理。

收到此命令后, 当前正在执行的任何 MoveTo、Move 或 Step 命令都应终止。收到 StopMoveStep 命令后, CurrentHue、EnhancedCurrentHue 和 CurrentSaturation 属性的值应保留为其当前值, 并且 RemainingTime 属性应设置为零。

#### 6.4.3.8.22 MoveColorTemperature 命令

MoveColorTemperature 命令允许光的色温以指定的速率变换。详见下表 212。

表 212 MoveColorTemperature 命令表

| 标识符 | 名称                            | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|-------------------------------|---------------|----------|------|----|-----|
| 0   | MoveMode                      | MoveModeEnum  | all      | —    | —  | M   |
| 1   | Rate                          | uint16        | all      | —    | —  | M   |
| 2   | ColorTemperatureMinimumMireds | uint16        | max65279 | —    | —  | M   |
| 3   | ColorTemperatureMaximumMireds | uint16        | max65279 | —    | —  | M   |
| 4   | OptionsMask                   | OptionsBitmap | desc     | —    | 0  | M   |
| 5   | OptionsOverride               | OptionsBitmap | desc     | —    | 0  | M   |

其中,

——MoveMode 字段

此字段应指示移动模式, 如 MoveHue 命令中所述。

——Rate 字段

此字段应以每秒步数为单位指示移动速率。一步是设备色温的一个单位变化。

——ColorTemperatureMinimumMireds 字段

此字段应指示当前移动操作的ColorTemperatureMireds属性的下限(色温的上限,单位为开尔文),以便:

ColorTempPhysicalMinMireds<=ColorTemperatureMinimumMireds字段<=ColorTemperatureMireds

因此,如果移动操作将ColorTemperatureMireds属性移向ColorTemperatureMinimumMireds字段的值,则应对其进行裁剪,以满足上述不变量。如果ColorTemperatureMinimumMireds字段设置为0,则应使用ColorTempPhysicalMinMireds作为ColorTemperatureMireds属性的下限。

——ColorTemperatureMaximumMireds 字段

此字段应指示当前移动操作的ColorTemperatureMireds属性的上限(以开尔文为单位的色温下限),以便:

ColorTemperatureMireds<=ColorTemperatureMaximumMireds字段<=ColorTempPhysicalMaxMireds

因此,如果移动操作将ColorTemperatureMireds属性移向ColorTemperatureMaximumMireds字段的值,则应对其进行裁剪,以便满足上述不变量。如果ColorTemperatureMaximumMireds字段设置为0,则应使用ColorTempPhysicalMaxMireds作为ColorTemperatureMireds属性的上限。

——OptionsMask 和 OptionsOverride 字段

这些字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

——收到指令的效果

收到此命令后,

- 如果MoveMode字段设置为1(向上)或3(向下),详见下表213。

- 如果Rate字段的值为零,则该命令无效,并且应返回响应,并将状态代码设置为INVALID\_COMMAND。

- 否则,设备应将ColorMode和EnhancedColorMode属性都设置为2(ColorTemperatureMireds)。

然后,设备应从其当前色温开始以连续的方式向上或向下移动,如表\*\*所述。

- 如果MoveMode字段设置为0(停止),则Rate字段应被完全忽略。

表213 MoveMode 字段表

| MoveMode | 收到命令后的动作                                                                                                                                                           |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Stop     | 如果移动,则停止运行,否则忽略该命令(即接受该命令,但没有任何作用)                                                                                                                                 |
| Up       | 按速率字段中给定的速率增加ColorTemperatureMireds属性(≡降低色温,单位开尔文)。如果ColorTemperatureMireds属性达到设备允许的峰值(通过ColorTemperatureMaximumMireds字段或ColorTempPhysicalMaxMireds属性),移动操作必须停止    |
| Down     | 按照Rate字段中给出的速率降低ColorTemperatureMireds属性(≡增加色温(开尔文))。如果ColorTemperatureMireds属性达到设备允许的最小值(通过ColorTemperatureMinimumMireds字段或ColorTempPhysicalMinMireds属性),则应停止移动操作 |

6.4.3.8.23 StepColorTemperature 命令

StepColor Temperature 命令允许光的色温以指定的步长进行步进,详见下表214。

表214 StepColor Temperature 命令表

| 标识符 | 名称                            | 类型            | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|-------------------------------|---------------|----------|------|----|-----|
| 0   | StepMode                      | StepModeEnum  | all      | –    | –  | M   |
| 1   | StepSize                      | uint16        | all      | –    | –  | M   |
| 2   | TransitionTime                | uint16        | max65534 | –    | –  | M   |
| 3   | ColorTemperatureMinimumMireds | uint16        | max65279 | –    | –  | M   |
| 4   | ColorTemperatureMaximumMireds | uint16        | max65279 | –    | –  | M   |
| 5   | OptionsMask                   | OptionsBitmap | desc     | –    | 0  | M   |
| 6   | OptionsOverride               | OptionsBitmap | desc     | –    | 0  | M   |

其中,

——StepMode 字段

此字段应指示要执行的步骤的模式，如StepHue命令中所述。

——StepSize 字段

该字段应指示要添加到设备色温当前值（或从中减去）的变化。

——TransitionTime 字段

此字段应以1/10秒为单位指示执行该步骤所需的时间。步骤是设备色温的变化，其幅度与StepSize字段相对应。

——ColorTemperatureMinimumMireds 字段

此字段应指示当前步骤操作的ColorTemperatureMireds属性的下限（色温的上限，单位为开尔文），以便：

ColorTempPhysicalMinMireds≤ColorTemperatureMinimumMireds字段≤ColorTemperatureMireds  
因此，如果步骤操作将ColorTemperatureMireds属性推向ColorTemperatureMinimumMireds字段的值，则应对其进行裁剪，以满足上述不变量。如果ColorTemperatureMinimumMireds字段设置为0，则应使用ColorTempPhysicalMinMireds作为ColorTemperatureMireds属性的下限。

——ColorTemperatureMaximumMireds 字段

此字段应指示当前步骤操作的ColorTemperatureMireds属性的上限（以开尔文为单位的色温下限），以便：

ColorTemperatureMireds≤ColorTemperatureMaximumMireds字段≤ColorTempPhysicalMaxMireds  
因此，如果步骤操作将ColorTemperatureMireds属性推向ColorTemperatureMaximumMireds字段的值，则应对其进行裁剪，以便满足上述不变量。如果ColorTemperatureMaximumMireds字段设置为0，则应使用ColorTempPhysicalMaxMireds作为ColorTemperatureMireds属性的上限。

——OptionsMask 和 OptionsOverride 字段

这些字段应根据OptionsMask和OptionsOverride字段的使用进行处理。

——收到指令的效果

收到此命令后，如果StepSize字段的值为零，则该命令无效，并且应返回响应，并将状态代码设置为INVALID\_COMMAND。

否则，设备应将ColorMode和EnhancedColorMode属性都设置为2（ColorTemperatureMireds）。然后，设备应从其当前色温向上或向下移动一步，如表215所述。

表215 StepMode 字段表

| StepMode | 收到命令后的动作                                                                                                                                                  |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| Up       | 将ColorTemperatureMireds属性增加一步（≡以开尔文为单位降低色温）。如果ColorTemperatureMireds属性达到设备允许的最大值（通过ColorTemperatureMaximumMireds字段或ColorTempPhysicalMaxMireds属性），则应停止步进操作 |
| Down     | 将ColorTemperatureMireds属性降低一步（≡增加色温（开尔文））。如果ColorTemperatureMireds属性达到设备允许的最小值（通过ColorTemperatureMinimumMireds字段或ColorTempPhysicalMinMireds属性），则应停止步进操作   |

6.4.4 镇流器配置功能集

6.4.4.1 概述

该功能集用于配置照明镇流器。

注：对镇流器配置功能集的支持是暂时的。

6.4.4.2 分类

镇流器配置功能集分类详见下表216。

表216 镇流器配置功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | BC      |

6.4.4.3 Cluster ID

镇流器配置功能集ID详见下表217。

表217 镇流器配置功能集 ID 表

| 标识符    | 名称                   | 一致性 |
|--------|----------------------|-----|
| 0x0301 | BallastConfiguration | P   |

6.4.4.4 依赖项

如果同一端点支持警报服务端群集，则启用警报功能，并且应支持LampAlarmMode属性。

6.4.4.5 数据类型

6.4.4.5.1 BallastStatusBitmap 类型

该数据类型源自map8，详见下表218。

表218 BallastStatusBitmap 类型

| Bit | 名称                    | 概述       | 一致性 |
|-----|-----------------------|----------|-----|
| 0   | BallastNonOperational | 镇流器的运行状态 | M   |
| 1   | LampFailure           | 灯的运行状态   | M   |

——BallastNonOperational 位  
该位应指示镇流器是否可运行。  
0=镇流器已完全运行  
1=镇流器未完全运行  
——LampFailure 位  
该位应指示所有灯是否正常运行。  
0=所有灯均可操作  
1=一个或多个灯未插入插座或有故障

6.4.4.5.2 LampAlarmModeBitmap 类型

该数据类型源自map8，详见下表219。

表219 LampAlarmModeBitmap 类型

| Bit | 名称            | 概述                  | 一致性 |
|-----|---------------|---------------------|-----|
| 0   | LampBurnHours | LampBurnHours状态警报生成 | M   |

其中，LampBurnHours位应表明LampBurnHours属性可以生成警报。

6.4.4.6 属性

镇流器配置功能集属性见下表220。

表220 镇流器配置功能集属性表

| 标识符    | 名称               | 类型                  | 约束条件                       | 质量等级 | 默认               | 通道   | 一致性 |
|--------|------------------|---------------------|----------------------------|------|------------------|------|-----|
| 0x0000 | PhysicalMinLevel | uint8               | 1to254                     | –    | 1                | RV   | M   |
| 0x0001 | PhysicalMaxLevel | uint8               | 1to254                     | –    | 254              | RV   | M   |
| 0x0002 | BallastStatus    | BallastStatusBitmap | all                        | –    | 0                | RV   | O   |
| 0x0010 | MinLevel         | uint8               | PhysicalMinLeveltoMaxLevel | –    | PhysicalMinLevel | RWVM | M   |
| 0x0011 | MaxLevel         | uint8               | MinLeveltoPhysicalMaxLevel | –    | PhysicalMaxLevel | RWVM | M   |
| 0x0012 | PowerOnLevel     | –                   |                            | –    | –                | –    | D   |
| 0x0013 | PowerOnFadeTime  | –                   |                            | –    | –                | –    | D   |

表220 镇流器配置功能集属性表（续）

| 标识符    | 名称                      | 类型                  | 约束条件    | 质量等级 | 默认   | 通道   | 一致性 |
|--------|-------------------------|---------------------|---------|------|------|------|-----|
| 0x0014 | IntrinsicBallastFactor  | uint8               | all     | X    | —    | RWVM | 0   |
| 0x0015 | BallastFactorAdjustment | uint8               | 100toMS | X    | null | RWVM | 0   |
| 0x0020 | LampQuantity            | uint8               | all     |      |      | RV   | M   |
| 0x0030 | LampType                | string              | max16   |      | “”   | RWVM | 0   |
| 0x0031 | LampManufacturer        | string              | max16   |      | “”   | RWVM | 0   |
| 0x0032 | LampRatedHours          | uint24              | all     | X    | null | RWVM | 0   |
| 0x0033 | LampBurnHours           | uint24              | all     | X    | 0    | RWVM | 0   |
| 0x0034 | LampAlarmMode           | LampAlarmModeBitmap | all     | —    | 0    | RWVM | 0   |
| 0x0035 | LampBurnHoursTripPoint  | uint24              | all     | X    | null | RWVM | 0   |

其中，

——PhysicalMinLevel 属性

此属性应根据调光曲线指定镇流器可实现的最小光输出。

——PhysicalMaxLevel 属性

此属性应根据调光曲线指定镇流器可实现的最大光输出。

——BallastStatus 属性

此属性应指定镇流器或所连接灯的各个方面的状态。

——MinLevel 属性

当级别控制功能集的CurrentLevel属性等于1（且开/关群集的OnOff属性等于TRUE）时，此属性应根据调光光曲线指定镇流器的光输出。

此属性的值应大于或等于PhysicalMinLevel且小于或等于MaxLevel。如果尝试将此属性设置为不满足这些条件的级别，则应返回响应，并将状态代码设置为CONSTRAINT\_ERROR，并且不应设置该级别。

——MaxLevel 属性

当级别控制功能集的CurrentLevel属性等于254（且开/关群集的OnOff属性等于TRUE）时，此属性应根据调光光曲线指定镇流器的光输出。

此属性的值应小于或等于PhysicalMaxLevel且大于或等于MinLevel。如果尝试将此属性设置为不满足这些条件的级别，则应返回响应，并将状态代码设置为CONSTRAINT\_ERROR，并且不应设置该级别。

——IntrinsicBallastFactor 属性

在任何调整之前，此属性应以百分比形式指定镇流器/灯组合的镇流器系数。

空值表示无效值。

——BallastFactorAdjustment 属性

此属性应以百分比形式指定应用于灯的配置光输出的倍增因子。此属性的典型用途是补偿灯使用寿命期间效率的降低。

光输出由下式给出：

实际光输出=配置光输出xBallastFactorAdjustment/100%

此属性的范围取决于制造商。如果尝试将此属性设置为无法支持的级别，则应返回响应，并将状态代码设置为CONSTRAINT\_ERROR，并且不得更改级别。空值表示未使用镇流器系数缩放。

——LampQuantity 属性

此属性应指定连接到此镇流器的灯的数量。

注：此数字不考虑灯是否实际位于插座中。

——LampType 属性

此属性应指定连接到镇流器的灯的类型（包括其瓦数）。

——LampManufacturer 属性

此属性应指定当前连接的灯的制造商名称。

——LampRatedHours 属性

此属性应指定制造商额定的灯的使用小时数。

空值表示时间无效或未知。

——LampBurnHours 属性

此属性应指定当前连接的灯自上次重新点亮以来累计的运行时间长度（以小时为单位）。如果灯已关闭，则不应累计运行时间。

更换灯时，应将此属性重置为零（例如远程重置）。如果连接了部分使用的灯，应更新LampBurnHours以反映灯的使用时间。

空值表示时间无效或未知。

——LampAlarmMode 属性

此属性应指定哪些属性可能导致生成警报通知。每个位位置上的1表示其相关属性能够生成警报。

注：注意所有警报也记录在警报表中。

——LampBurnHoursTripPoint 属性

此属性应指定在生成警报之前LampBurnHours属性可以达到的小时数。

如果同一设备上不存在警报功能集，则不使用此属性，因此可以省略（参见章节“6.4.4.4 依赖项”）。

生成的警报中包含的警报代码字段应为0x01。如果该属性的值为空，则不会生成此警报。

6.4.4.7 调光光变曲线

调光曲线建议采用对数曲线，定义如下公式（9）：

$$\%Light = \dots\dots\dots (9)$$

式中：%Light 是镇流器的光输出百分比，且 Level 是一个 8 位整数，介于 1（0.1%光输出）和 254（100%输出）之间，使用以下公式（10）调整 MinLevel 和 MaxLevel：

$$level = * + /253 \dots\dots\dots (10)$$

注1：值 255 未被使用。

注2：光输出由此曲线与IntrinsicBallastFactor和BallastFactorAdjustment属性共同决定。下表221给出了不同MinLevel、MaxLevel和 CurrentLevel值的调光灯光曲线的几个示例。

表221 调光灯光曲线示例表

| MinLevel | MaxLevel | CurrentLevel | Level | %Light |
|----------|----------|--------------|-------|--------|
| 1        | 254      | 1            | 1     | 0.10%  |
| 1        | 254      | 10           | 10    | 0.13%  |
| 1        | 254      | 100          | 100   | 1.49%  |
| 1        | 254      | 254          | 254   | 100%   |
| 170      | 254      | 254          | 254   | 100%   |
| 170      | 230      | 1            | 170   | 10.1%  |
| 170      | 230      | 10           | 172   | 10.7%  |
| 170      | 230      | 100          | 193   | 19.2%  |
| 170      | 230      | 254          | 230   | 51.9%  |

6.5 暖通空调

6.5.1 概述

功能集库由单独的章节组成，例如本章。章节之间的引用使用X.Y表示法进行，其中X是章节，Y是该章节中的子节。

6.5.2 暖通空调功能集列表

6.5.2.1 概述

本文档中定义的功能集通常用于暖通空调的应用程序，但可以在任何应用程序域中使用。

6.5.2.2 术语

预冷：在一天中较早（较凉爽）的时刻冷却建筑物，这样一来，比起在一天中较晚（较热）冷却所需的热容更少。

6.5.2.3 Cluster ID

本节列出了本章中指定的HVAC特定功能集ID表，详见下表222。

表222 HVAC 功能集 ID 表

| 功能集标识符 | 功能集名称     | 描述                    |
|--------|-----------|-----------------------|
| 0x0200 | 热泵配置和控制   | 用于配置和控制泵的接口           |
| 0x0201 | 恒温器       | 用于配置和控制恒温器功能的接口       |
| 0x0202 | 风扇控制      | 用于控制风扇的接口             |
| 0x0204 | 恒温器用户界面配置 | 配置恒温器（可对恒温器远程）用户界面的接口 |
| 0x0081 | 阀门配置和控制   | 配置和控制阀门功能的接口          |

6.5.3 恒温功能集

6.5.3.1 概述

该功能集提供了一个接口，用于访问和控制恒温器的功能。如下图19所示。

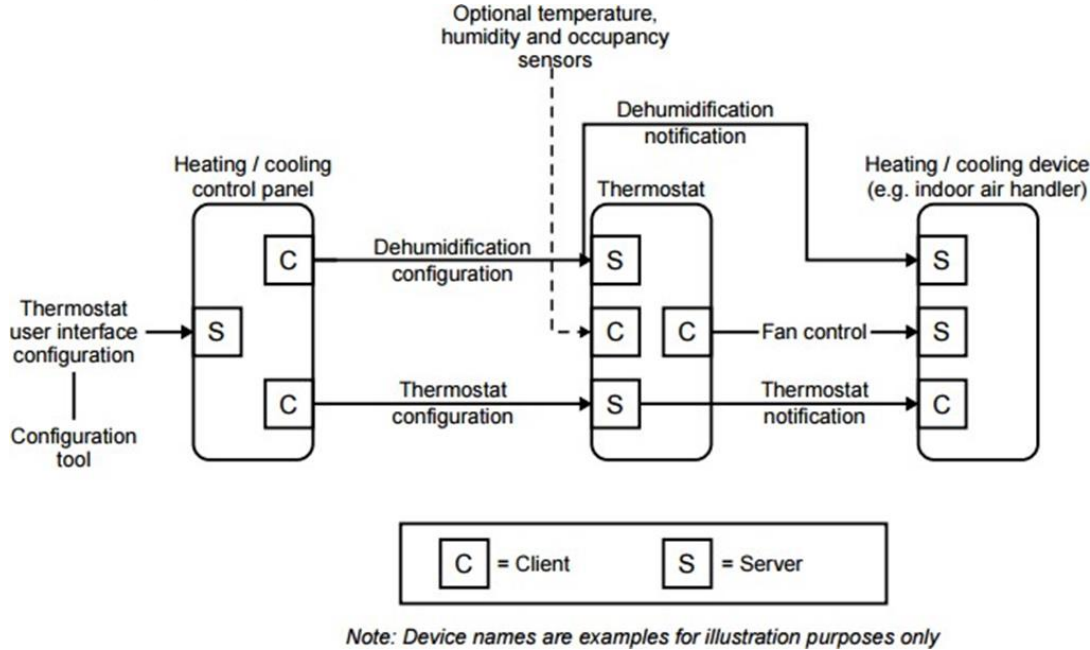


图19 恒温器及相关功能集的示例用法

6.5.3.2 分类

恒温功能集分类详见下表223。

表223 恒温功能集 ID 表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | TSTAT   |

6.5.3.3 Cluster ID

恒温功能集ID详见下表224。

表224 恒温功能集 ID 表

| 标识符    | 名称               |
|--------|------------------|
| 0x0201 | 恒温器 (Thermostat) |

#### 6.5.3.4 功能属性

恒温功能集应支持以下定义的Feature位图功能属性，详见下表225。

表225 恒温功能集属性表

| Bit | 编码   | 特征                               | 一致性         | 概述                 |
|-----|------|----------------------------------|-------------|--------------------|
| 0   | HEAT | Heating                          | AUTO, 0.a+  | 恒温器支持控制加热系统        |
| 1   | COOL | Cooling                          | AUTO, 0.a+  | 恒温器支持控制冷却系统        |
| 2   | OCC  | Occupancy                        | 0           | 支持有人和无人调整设备设定值     |
| 3   | SCH  | ScheduleConfigu-ration           | [Zigbee], D | 支持远程设定和调整不同时间段的设定值 |
| 4   | SB   | Setback                          | 0           | 支持自定义回调设定值（或变化范围）  |
| 5   | AUTO | AutoMode                         | 0           | 支持系统自动模式           |
| 6   | LTNE | LocalTempera-tureNotExposed      | 0           | 恒温器本地温度数据不对外部上报    |
| 7   | MSCH | MSCHMatterSchedule-Configuration | 0           | 支持高级日程配置           |
| 8   | PRES | Presets                          | 0           | 支持预设温度值            |

其中，LocalTemperatureNotExposed 功能表示内部使用的计算后的本地温度（CalculatedLocalTemperature）无法对外部上报，例如由于温度控制由一个单独的子系统完成，该子系统不提供当前测量温度的视图，但允许设定设定点。

#### 6.5.3.5 温度单位

本功能集中的温度由使用摄氏度（degree Celsius）为单位的类型表示。

本功能集中使用的温度数据类型在数据模型的派生数据类型部分有定义。以下与温度相关的数据类型也在该功能集中定义并使用：

- TemperatureDifference
- SignedTemperature
- UnsignedTemperature

虽然温度值必须使用这些类型进行无线传输，但并不限制恒温器如何显示或存储温度值。

注：温度属性的计算在进行计算或比较时，属性值必须转换为共同的类型。在许多情况下，仅仅使用整数表示是不够的，因为从摄氏度到整数值的缩放比例是不同的。

#### 6.5.3.6 设定点限制

若干属性对设定点值施加了限制，这意味着任何修改限制或设定点的机制都必须维持这些约束。各属性描述中详细说明了在修改值时发生冲突应采取的措施。

用户可配置的限制必须在设备限制范围内：

•  

$$\text{AbsMinHeatSetpointLimit} \leq \text{MinHeatSetpointLimit} \leq \text{MaxHeatSetpointLimit} \leq \text{AbsMaxHeatSetpointLimit}$$

•  

$$\text{AbsMinCoolSetpointLimit} \leq \text{MinCoolSetpointLimit} \leq \text{MaxCoolSetpointLimit} \leq \text{AbsMaxCoolSetpointLimit}$$

设定点必须在用户可配置的限制范围内：

- $\text{MinHeatSetpointLimit} \leq \text{OccupiedHeatingSetpoint} \leq \text{MaxHeatSetpointLimit}$
- $\text{MinCoolSetpointLimit} \leq \text{OccupiedCoolingSetpoint} \leq \text{MaxCoolSetpointLimit}$
- $\text{MinHeatSetpointLimit} \leq \text{UnoccupiedHeatingSetpoint} \leq \text{MaxHeatSetpointLimit}$

- $\text{MinCoolSetpointLimit} \leq \text{UnoccupiedCoolingSetpoint} \leq \text{MaxCoolSetpointLimit}$   
仅当支持AUTO功能时，加热和冷却设定点及限制之间必须保持中性区（deadband）：
- $\text{AbsMinHeatSetpointLimit} \leq (\text{AbsMinCoolSetpointLimit} - \text{MinSetpointDeadBand})$
- $\text{AbsMaxHeatSetpointLimit} \leq (\text{AbsMaxCoolSetpointLimit} - \text{MinSetpointDeadBand})$
- $\text{MinHeatSetpointLimit} \leq (\text{MinCoolSetpointLimit} - \text{MinSetpointDeadBand})$
- $\text{MaxHeatSetpointLimit} \leq (\text{MaxCoolSetpointLimit} - \text{MinSetpointDeadBand})$
- $\text{OccupiedHeatingSetpoint} \leq (\text{OccupiedCoolingSetpoint} - \text{MinSetpointDeadBand})$
- $\text{UnoccupiedHeatingSetpoint} \leq (\text{UnoccupiedCoolingSetpoint} - \text{MinSetpointDeadBand})$

#### 6.5.3.7 依赖关系

如果同一端点上支持Alarmsserver功能集，则启用Alarms功能，并且必须支持AlarmMask属性。

对于远程温度感应，可以在同一端点上包含TemperatureMeasurementclient功能集。对于人体感应，可以在同一端点上包含OccupancySensingclient功能集。

#### 6.5.3.8 数据类型

##### 6.5.3.8.1 TemperatureDifference 类型

此数据类型派生自int16，用于表示温度差，分辨率为0.01℃。

- $\text{value} = (\text{temperaturein}^\circ\text{C}) \times 100$
- $-4^\circ\text{C} \rightarrow -400$
- $123.45^\circ\text{C} \rightarrow 12345$

可使用的完整（non-null）范围，即从-327.67℃到327.67℃。

##### 6.5.3.8.2 SignedTemperature 类型

此数据类型派生自int8，用于表示从-12.7℃到12.7℃的温度，分辨率为0.1℃。

- $\text{value} = (\text{temperaturein}^\circ\text{C}) \times 10$
- $-4^\circ\text{C} \rightarrow -40$
- $12.3^\circ\text{C} \rightarrow 123$

此类型适用于需要紧凑表示，并且在分辨率和范围上仍能满足需要的场景。

##### 6.5.3.8.3 UnsignedTemperature 类型

此数据类型派生自unit8，用于表示从0℃到25.5℃的温度，分辨率为0.1℃。

- $\text{value} = (\text{temperaturein}^\circ\text{C}) \times 10$
- $12.3^\circ\text{C} \rightarrow 123$

此类型适用于需要紧凑表示，并且在分辨率和范围上仍能满足需要的场景。

##### 6.5.3.8.4 ALErrorCodeBitmap 类型

该数据类型派生自map32，详见下表226。

表226 ALError CodeBitmap 类型

| Bit | 名称                | 概述          | 一致性 |
|-----|-------------------|-------------|-----|
| 0   | CompressorFail    | 压缩机故障或冷却剂泄露 | M   |
| 1   | RoomSensorFail    | 室内温度传感器故障   | M   |
| 2   | OutdoorSensorFail | 室外温度传感器故障   | M   |
| 3   | CoilSensorFail    | 室内盘管温度传感器故障 | M   |
| 4   | FanFail           | 风扇故障        | M   |

##### 6.5.3.8.5 AlarmCodeBitmap 类型

该数据类型派生自map8，详见下表227。

表227 AlarmCodeBitmap 类型

| Bit | 名称              | 概述                  | 一致性 |
|-----|-----------------|---------------------|-----|
| 0   | Initialization  | 初始化失败。设备在上电时未能完成初始化 | M   |
| 1   | Hardware        | 硬件故障                | M   |
| 2   | SelfCalibration | 自校准失败               | M   |

#### 6.5.3.8.6 HVACSystemTypeBitmap 类型

该数据类型派生自map8，详见下表228。

表228 HVACSystemTypeBitmap 类型

| Bit  | 名称                | 概述              | 一致性 |
|------|-------------------|-----------------|-----|
| 1..0 | CoolingStage      | HVAC系统正在使用的冷却阶段 | M   |
| 3..2 | HeatingStage      | HVAC系统正在使用的加热阶段 | M   |
| 4    | HeatingIsHeatPump | 加热类型是否为热泵       | M   |
| 5    | HeatingUsesFuel   | HVAC系统是否使用燃料    | M   |

——CoolingStageBits 位

这些位应指示HVAC系统正在使用的冷却阶段。

- 00=CoolStage1
- 01=CoolStage2
- 10=CoolStage3
- 11=Reserved

——HeatingStageBits 位

这些位应指示HVAC系统正在使用的加热阶段。

- 00=HeatStage1
- 01=HeatStage2
- 10=HeatStage3
- 11=Reserved

——HeatingIsHeatPump 位

此位应指示HVAC系统是传统系统还是热泵。

- 0=Conventional
- 1=HeatPump

——HeatingUsesFuel 位

此位应指示HVAC系统是否使用燃料。

0=Doesnotusefuel

1=Usesfuel

#### 6.5.3.8.7 OccupancyBitmap 类型

OccupancyBitmap 数据类型的详细信息见表229。

表229 OccupancyBitmap 类型表

| Bit | 名称       | 概述     | 一致性 |
|-----|----------|--------|-----|
| 0   | Occupied | 指示占用状态 | M   |

如果设置了Occupied位，应指示占用状态，否则如果未设置该位，应指示未占用状态。

#### 6.5.3.8.8 PresetTypeFeaturesBitmap 类型

该数据类型派生自map16，详见下表230。

表230 PresetTypeFeaturesBitmap 类型

| Bit | 名称            | 概述           | 一致性 |
|-----|---------------|--------------|-----|
| 0   | Automatic     | 预设可能由恒温器自动激活 | M   |
| 1   | SupportsNames | 预设支持用户提供的名称  | M   |

## 6.5.3.8.9 ProgrammingOperationModeBitmap 类型

该数据类型派生自map8，详见下表231。

表231 ProgrammingOperationModeBitmap 类型

| Bit | 名称             | 概述                    | 一致性 |
|-----|----------------|-----------------------|-----|
| 0   | ScheduleActive | 计划编程模式。这支持任何预设的每周计划配置 | M   |
| 1   | AutoRecovery   | 自动/恢复模式               | M   |
| 2   | Economy        | 经济/能源之星模式             | M   |

## 6.5.3.8.10 RelayStateBitmap 类型

该数据类型派生自map16，详见下表232。

表232 RelayStateBitmap 类型

| Bit | 名称         | 概述             | 一致性 |
|-----|------------|----------------|-----|
| 0   | Heat       | HeatStage0n    | M   |
| 1   | Cool       | CoolStage0n    | M   |
| 2   | Fan        | FanStage0n     | M   |
| 3   | HeatStage2 | Heat2ndStage0n | M   |
| 4   | CoolStage2 | Cool2ndStage0n | M   |
| 5   | FanStage2  | Fan2ndStage0n  | M   |
| 6   | FanStage3  | Fan3rdStage0n  | M   |

## 6.5.3.8.11 RemoteSensingBitmap 类型

该数据类型派生自map8，详见下表233。

表233 RemoteSensingBitmap 类型

| Bit | 名称                 | 概述                               | 一致性  |
|-----|--------------------|----------------------------------|------|
| 0   | LocalTemperature   | CalculatedLocalTemperature源自远程节点 | M    |
| 1   | OutdoorTemperature | OutdoorTemperature来源于远程节点        | desc |
| 2   | Occupancy          | Occupancy源自远程节点                  | OCC  |

如果支持OutdoorTemperature属性，则OutdoorTemperature位应当被支持。

## 6.5.3.8.12 ScheduleTypeFeaturesBitma 类型

该数据类型派生自map16，详见下表234。

表234 ScheduleTypeFeaturesBitma 类型

| Bit | 名称                | 概述                 | 一致性        |
|-----|-------------------|--------------------|------------|
| 0   | SupportsPresets   | 支持预设               | [PRES]. b+ |
| 1   | SupportsSetpoints | 支持设定值              | 0. b+      |
| 2   | SupportsNames     | 支持用户提供的名称          | 0          |
| 3   | SupportsOff       | 支持转换到SystemModeOff | 0          |

其中，

- SupportsPresets位：该位应当指示，任何具有与包含ScheduleTypeStruct的SystemMode字段值相匹配的SystemMode字段的ScheduleStruct，都支持在其Transitions字段中包含的ScheduleTransitionStructs上指定预设。

- SupportsSetpoints位: 该位应当指示,任何具有与包含ScheduleTypeStruct的SystemMode字段值相匹配的SystemMode字段的ScheduleStruct, 都支持在其Transitions字段中包含的ScheduleTransitionStructs上指定预设点。
- SupportsNames位: 该位应当指示,任何具有与包含ScheduleTypeStruct的SystemMode字段值相匹配的SystemMode字段的ScheduleStruct, 都支持设置Name字段的值。
- SupportsNames位: 该位应当指示,任何具有与包含ScheduleTypeStruct的SystemMode字段值相匹配的SystemMode字段的ScheduleStruct, 都支持将其SystemMode字段设置为关闭。

#### 6.5.3.8.13 ScheduleDayOfWeekBitmap 类型

该数据类型派生自map8, 详见下表235。

表235 ScheduleDayOfWeekBitmap 类型

| Bit | 名称        | 概述    | 一致性 |
|-----|-----------|-------|-----|
| 0   | Sunday    | 星期日   | M   |
| 1   | Monday    | 星期一   | M   |
| 2   | Tuesday   | 星期二   | M   |
| 3   | Wednesday | 星期三   | M   |
| 4   | Thursday  | 星期四   | M   |
| 5   | Friday    | 星期五   | M   |
| 6   | Saturday  | 星期六   | M   |
| 7   | Away      | 外出或度假 | M   |

#### 6.5.3.8.14 ScheduleModeBitmap 类型

该数据类型派生自map8, 详见下表236。

表236 ScheduleModeBitmap 类型

| Bit | 名称                  | 概述      | 一致性 |
|-----|---------------------|---------|-----|
| 0   | HeatSetpointPresent | 调整加热设定点 | M   |
| 1   | CoolSetpointPresent | 调整冷却设定点 | M   |

#### 6.5.3.8.15 ACCapacityFormatEnum 类型

该数据类型派生自enum8, 详见下表237。

表237 ACCapacityFormatEnum 类型

| Bit | 名称   | 概述        | 一致性 |
|-----|------|-----------|-----|
| 0   | BTUh | 英制热量单位/小时 | 0   |

#### 6.5.3.8.16 ACCompressorTypeEnum 类型

该数据类型派生自enum8, 详见下表238。

表238 ACCompressor TypeEnum 类型

| Bit | 名称      | 概述        | 一致性 |
|-----|---------|-----------|-----|
| 0   | Unknown | 未知压缩机类型   | 0   |
| 1   | T1      | 最高工作环境43℃ | 0   |
| 2   | T2      | 最高工作环境35℃ | 0   |
| 3   | T3      | 最高工作环境52℃ | 0   |

#### 6.5.3.8.17 ACLouverPositionEnum 类型

该数据类型派生自enum8, 详见下表239。

表239 ACLouverPositionEnum 类型

| Bit | 名称            | 概述     | 一致性 |
|-----|---------------|--------|-----|
| 1   | Closed        | 完全关闭   | 0   |
| 2   | Open          | 完全开启   | 0   |
| 3   | Quarter       | 四分之一开启 | 0   |
| 4   | Half          | 半开启    | 0   |
| 5   | ThreeQuarters | 四分之三开启 | 0   |

## 6.5.3.8.18 ACRefrigerantTypeEnum 类型

该数据类型派生自enum8，详见下表240。

表240 ACRefrigerantTypeEnum 类型

| Bit | 名称      | 概述       | 一致性 |
|-----|---------|----------|-----|
| 0   | Unknown | 未知制冷剂类型  | 0   |
| 1   | R22     | R22制冷剂   | 0   |
| 2   | R410a   | R410a制冷剂 | 0   |
| 3   | R407c   | R407c制冷剂 | 0   |

## 6.5.3.8.19 ACTYPEEnum 类型

该数据类型派生自enum8，详见下表241。

表241 ACTYPEEnum 类型

| Bit | 名称               | 概述     | 一致性 |
|-----|------------------|--------|-----|
| 0   | Unknown          | 未知AC类型 | 0   |
| 1   | CoolingFixed     | 冷却和定速  | 0   |
| 2   | HeatPumpFixed    | 热泵和定速  | 0   |
| 3   | CoolingInverter  | 冷却和变频器 | 0   |
| 4   | HeatPumpInverter | 热泵和变频  | 0   |

## 6.5.3.8.20 SetpointRaiseLowerModeEnum 类型

该数据类型派生自enum8，详见下表242。

表242 SetpointRaiseLowerModeEnum 类型

| Bit | 名称   | 概述         | 一致性       |
|-----|------|------------|-----------|
| 0   | Heat | 调整加热设定点    | HEAT      |
| 1   | Cool | 调整冷却设定点    | COOL      |
| 2   | Both | 调整加热和冷却设定点 | HEAT COOL |

## 6.5.3.8.21 ControlSequenceOfOperationEnum 类型

该数据类型派生自enum8，详见下表243。

表243 ControlSequenceOfOperationEnum 类型

| Bit | 名称                           | 概述           | 一致性         |
|-----|------------------------------|--------------|-------------|
| 0   | CoolingOnly                  | 加热和紧急情况是不可用的 | [COOL]      |
| 1   | CoolingWithReheat            | 加热和紧急情况是不可用的 | [COOL]      |
| 2   | HeatingOnly                  | 冷却和预冷却是不可用的  | [HEAT]      |
| 3   | HeatingWithReheat            | 冷却和预冷却是不可用的  | [HEAT]      |
| 4   | CoolingAndHeating            | 所有模式都可用      | [HEAT&COOL] |
| 5   | CoolingAndHeating-WithReheat | 所有模式都可用      | [HEAT&COOL] |

其中，一个表示支持CoolingAndHeating（或CoolingAndHeatingWithReheat）的恒温器应该能够按需请求加热或冷却，并且通常会支持AutoSystemMode。

支持冷却或加热的系统，应该根据当前能力报告CoolingOnly或者HeatingOnly，并且需要外部干预才能改变模式，或者整个构造必须处于同一模式。

#### 6.5.3.8.22 PresetScenarioEnum 类型

该数据类型派生自enum8，详见下表244。

表244 PresetScenarioEnum 类型

| 值   | 名称           | 概述         | 一致性 |
|-----|--------------|------------|-----|
| 1   | Occupied     | 恒温器控制的区域有人 | M   |
| 2   | Unoccupied   | 恒温器控制的区域无人 | M   |
| 3   | Sleep        | 用户可能在睡觉    | M   |
| 4   | Wake         | 用户可能正在醒来   | M   |
| 5   | Vacation     | 用户正在度假     | M   |
| 6   | GoingToSleep | 用户可能正在去睡觉  | M   |
| 254 | UserDefined  | 自定义预设      | M   |

其中，

- Undefined值没有指定的场景。
- Occupied值指示恒温器的温度控制区域被占用期间的预设。适用于可以自动确定占用情况的恒温器。
- Unoccupied值指示恒温器的温度控制区域未被占用期间的预设。适用于可以自动确定占用情况的恒温器。
- Sleep值指示用户可能正在睡觉期间的预设。
- Wake值指示用户可能正在醒来期间的预设。
- Vacation值指示用户度假或长时间不在家期间的预设。
- GoingToSleep值指示用户可能要睡觉期间的预设。
- UserDefined值指示一个自由形式的预设；设置后，PresetStruct上的Name字段不应为空。

#### 6.5.3.8.23 SetpointChangeSourceEnum 类型

该数据类型派生自enum8，详见下表245。

表245 SetpointChangeSourceEnum 类型

| 值 | 名称       | 概述                           | 一致性        |
|---|----------|------------------------------|------------|
| 0 | Manual   | 手动，用户通过恒温器发起的设定点更改           | 0          |
| 1 | Schedule | 计划/内部编程引发的设定点改变              | [SCH MSCH] |
| 2 | External | 外部发起的设定点更改（例如，DRLC群集命令，属性写入） | 0          |

#### 6.5.3.8.24 StartOfWeekEnum 类型

该数据类型派生自enum8，详见下表246。

表246 StartOfWeekEnum 类型

| 值 | 名称        | 概述 | 一致性 |
|---|-----------|----|-----|
| 0 | Sunday    | —  | M   |
| 1 | Monday    | —  | M   |
| 2 | Tuesday   | —  | M   |
| 3 | Wednesday | —  | M   |
| 4 | Thursday  | —  | M   |
| 5 | Friday    | —  | M   |
| 6 | Saturday  | —  | M   |

#### 6.5.3.8.25 SystemModeEnum 类型

该数据类型派生自enum8，详见下表247以及表248。

表247 SystemModeEnum 类型

| 值 | 名称            | 概述              | 一致性    |
|---|---------------|-----------------|--------|
| 0 | Off           | 恒温器不产生冷却或加热需求   | 0      |
| 1 | Auto          | 根据需要产生冷却或加热需求   | AUTO   |
| 3 | Cool          | 仅产生冷却需求         | [COOL] |
| 4 | Heat          | 一般加热需求          | [HEAT] |
| 5 | EmergencyHeat | 使用第二阶段加热来达到所需温度 | [HEAT] |
| 6 | Precooling    | (seeTerms)      | [COOL] |
| 7 | FanOnly       | —               | 0      |
| 8 | Dry           | —               | 0      |
| 9 | Sleep         | —               | 0      |

表248 解释加热、冷却和自动 SystemModeEnum 值

| AttributeValues | TemperatureBelowHeatSetpoint | TemperatureBetweenHeatSetpointandCoolSetpoint | TemperatureAboveCoolSetpoint |
|-----------------|------------------------------|-----------------------------------------------|------------------------------|
| Heat            | 温度低于目标                       | 温度达到目标                                        | 温度达到目标                       |
| Cool            | 温度达到目标                       | 温度达到目标                                        | 温度高于目标                       |
| Auto            | 温度低于目标                       | 温度达到目标                                        | 温度高于目标                       |

6.5.3.8.26 ThermostatRunningModeEnum 类型

该数据类型派生自enum8，详见下表249。

表249 ThermostatRunningModeEnum 类型

| 值 | 名称   | 概述            | 一致性    |
|---|------|---------------|--------|
| 0 | Off  | 恒温器不产生冷却或加热需求 | 0      |
| 3 | Cool | 仅产生冷却需求       | [COOL] |
| 4 | Heat | 仅产生加热需求       | [HEAT] |

6.5.3.8.27 TemperatureSetpointHoldEnum 类型

该数据类型派生自enum8，详见下表250。

表250 TemperatureSetpointHoldEnum 类型

| 值 | 名称              | 概述             | 一致性 |
|---|-----------------|----------------|-----|
| 0 | SetpointHoldOff | 按照调度程序         | M   |
| 1 | SetpointHoldOn  | 保持当前设定点，忽略调度程序 | M   |

6.5.3.8.28 PresetStruct 类型

PresetStruct 类型描述详见下表251。

表251 PresetStruct 类型

| 标识符 | 名称              | 类型                 | 约束条件  | 质量等级 | 默认    | 通道 | 一致性  |
|-----|-----------------|--------------------|-------|------|-------|----|------|
| 0   | PresetHandle    | octstr             | max16 | X    | —     | —  | M    |
| 1   | PresetScenario  | PresetScenarioEnum | all   | —    | —     | —  | M    |
| 2   | Name            | string             | max64 | X    | null  | —  | 0    |
| 3   | CoolingSetpoint | temperature        | desc  | —    | 26°C  | —  | COOL |
| 4   | HeatingSetpoint | temperature        | desc  | —    | 20°C  | —  | HEAT |
| 5   | BuiltIn         | bool               | all   | —    | false | —  | M    |

其中，

——PresetHandle 字段

此字段应指示为此预设生成的设备标识符。在设备上，它应是唯一的，在与之关联的预设被删除后，不应重复使用。

此字段仅在将包含PresetStruct的Presets属性增加以创建新Presets的目的时为null。有关Preset处理的创建，请参阅“6.6.3.10 属性”中的Presets描述。

——PresetScenario 字段

此字段选择与此预设关联的PresetScenarioEnum值。

——Name 字段

此字段应指示用户提供的名称。null值表示没有名称。在具有相同PresetScenario字段值的每个Preset子集内，Name字段中不应有任何具有相同值（包括null值）的Preset。

——CoolingSetpoint 字段

此字段指示所预设的冷却设定点。参见“6.6.3.7 设定点限制”以了解约束。

——HeatingSetpoint 字段

此字段指示所预设的加热设定点。参见“6.6.3.7 设定点限制”以了解约束。

——BuiltIn 字段

此字段指示预设是否标记为“内置”，这意味着它可以被修改，但不能被删除。

#### 6.5.3.8.29 PresetTypeStruct 类型

PresetTypeStruct 类型描述详见下表252。

表252 PresetTypeStruct 类型

| 标识符 | 名称                 | 类型                       | 约束条件 | 质量等级 | 默认 | 通道 | 一致性 |
|-----|--------------------|--------------------------|------|------|----|----|-----|
| 0   | PresetScenario     | PresetScenarioEnum       | all  | —    | —  | —  | M   |
| 1   | NumberOfPresets    | uint8                    | all  | —    | 0  | —  | M   |
| 2   | PresetTypeFeatures | PresetTypeFeaturesBitmap | all  | —    | 0  | —  | M   |

其中，

- PresetScenario字段应指定此恒温器支持的PresetScenarioEnum值。
- NumberOfPresets字段应指定此PresetScenarioEnum的预设数量限制。
- PresetTypeFeatures字段应指定此PresetTypeStruct的特性位图。

#### 6.5.3.8.30 WeeklyScheduleTransitionStruct 类型

这代表恒温器时间表中的单个转换，详见下表253。

表253 WeeklyScheduleTransitionStruct 类型

| 标识符 | 名称             | 类型          | 约束条件    | 质量等级 | 默认 | 通道 | 一致性 |
|-----|----------------|-------------|---------|------|----|----|-----|
| 0   | TransitionTime | uint16      | max1439 | —    | —  | —  | M   |
| 1   | HeatSetpoint   | temperature | all     | X    | —  | —  | M   |
| 2   | CoolSetpoint   | temperature | all     | X    | —  | —  | M   |

其中，

- TransitionTime字段应表示在相关天的计划转换开始时间。时间将由一个16位无符号整数表示，以指定自午夜以来的分钟数。例如，上午6点将由自午夜以来的360分钟表示，晚上11:30将由自午夜以来的1410分钟表示。
- HeatSetpoint字段应表示在此相关转换开始时间应用的加热设定点。
- CoolSetpoint字段应表示与此关联的转换开始时间应用的冷却设定值。

#### 6.5.3.8.31 ScheduleStruct 类型

ScheduleStruct 类型描述详见下表254。

表254 ScheduleStruct 类型

| 标识符 | 名称             | 类型                             | 约束条件                           | 质量等级 | 默认    | 通道 | 一致性 |
|-----|----------------|--------------------------------|--------------------------------|------|-------|----|-----|
| 0   | ScheduleHandle | octstr                         | max16                          | X    | -     | -  | M   |
| 1   | SystemMode     | SystemModeEnum                 | desc                           | -    | -     | -  | M   |
| 2   | Name           | string                         | max64                          | -    | -     | -  | O   |
| 3   | PresetHandle   | octstr                         | max16                          | -    | -     | -  | O   |
| 4   | Transitions    | list[ScheduleTransitionStruct] | ltoNumberOfScheduleTransitions | -    | empty | -  | M   |
| 5   | BuiltIn        | bool                           | all                            | X    | false | -  | M   |

其中，

——ScheduleHandle 字段

此字段应指示此计划的设备生成的标识符。在设备上应是唯一的，并且在关联的计划被删除后不应重复使用。

当包含的ScheduleStruct被附加到Schedules属性以创建新计划时，此字段仅应为null。有关创建Schedule的内容，请参阅“6.6.3.10 Schedules”。

——SystemMode 字段

此字段应指定此计划中transitions的默认恒温器系统模式。此字段的有效值仅应为Auto、Heat和Cool。

——Name 字段

此字段应指定ScheduleStruct的名称。

——PresetHandle 字段

此字段应指示此计划中Transitions的默认PresetHandle值。

——Transitions 字段

此字段应指定时间表的状态转换列表。

此字段不应包含具有相同TransitionTime字段和重叠DayOfWeek字段的多个ScheduleStruct；即不应有重复的状态转换。

如果NumberOfScheduleTransitionsPerDay属性不为空，则对于ScheduleDayOfWeekBitmap中的每个位具有关联在DayOfWeek中设置的转换次数不应大于NumberOfScheduleTransitionsPerDay属性的值。

为了确定此列表中当前活动的ScheduleStruct，当前时间应为当前时间显示值中午夜后的分钟数，而不是自午夜以来实际经过的分钟数。在转换至或从夏令时的日期中，转换期间将某些值可能会重复出现或完全不出现。

此列表中的ScheduleTransitionStruct如果其DayOfWeek字段与当前星期几匹配，并且当前时间大于等于TransitionTime，但小于下个Transitions字段中任何其他ScheduleTransitionStruct的TransitionTime（其DayOfWeek字段也与当前星期几匹配），则应处于活跃状态。

如果当前时间小于所有DayOfWeek字段与当前星期几匹配的ScheduleTransitionStruct，服务端应当将当前时间视为在相同日期的过去的计划转换时间。ScheduleTransitionStruct通过比较当前时间与计划转换时间来确定当前活跃的ScheduleTransitionStruct或者当前日期为当前星期几。如果没有找到有效的ScheduleTransitionStruct，则符合条件的ScheduleStruct应为DayOfWeek字段与当前星期几相同且相符的ScheduleTransitionStruct集合中TransitionTime最后一个ScheduleTransitionStruct。

——BuiltIn 字段

此字段应指示预设是否标记为“内置”，这意味着它可以被修改，但不能被删除。

6.5.3.8.32 ScheduleTransitionStruct 类型

ScheduleTransitionStruct 类型描述详见下表255。

表255 ScheduleTransitionStruct 类型

| 标识符 | 名称              | 类型                      | 约束条件    | 质量等级 | 默认 | 通道 | 一致性    |
|-----|-----------------|-------------------------|---------|------|----|----|--------|
| 0   | DayOfWeek       | ScheduleDayOfWeekBitmap | desc    | —    | —  | —  | M      |
| 1   | TransitionTime  | uint16                  | max1439 | —    | —  | —  | M      |
| 2   | PresetHandle    | octstr                  | max16   | —    | —  | —  | [PRES] |
| 3   | SystemMode      | SystemModeEnum          | desc    | —    | —  | —  | 0      |
| 4   | CoolingSetpoint | temperature             | desc    | —    | —  | —  | [COOL] |
| 5   | HeatingSetpoint | temperature             | desc    | —    | —  | —  | [HEAT] |

此结构提供了一天中的时间和一周中的几天，用于时间表中的状态转换。恒温器应在转换时间确定新设定点时使用以下优先级顺序：

- a) 如果提供了 PresetHandle 字段，则应使用具有该标识符的 Presets 属性中的 PresetStruct 的设定点
- b) 如果提供了 HeatingSetpoint 或 CoolingSetpoint，则应使用它们。
  - 1) 如果提供了 SystemMode 字段，则使用 SystemMode 字段来解释 HeatingSetpoint 和 CoolingSetpoint 字段。
  - 2) 如果未提供 SystemMode 字段，则应使用父 ScheduleStruct 上的 SystemMode 字段来解释 HeatingSetpoint 和 CoolingSetpoint 字段。
- c) 如果未提供 PresetHandle 字段或任何 Setpoint 字段，则应使用父 ScheduleStruct 上的 PresetHandle 字段来确定活动的 PresetStruct。
- d) 如果未提供 PresetHandle 且未为当前 SystemMode 提供设定点，则服务端应使用当前 SystemMode 的默认值。

如果设定点源自预设，则 ActivePresetHandle 应设置为该预设的 PresetHandle。如果使用 CoolingSetpoint 来确定冷却设定点：

- a) 如果服务端支持 OCC 功能，并且 Occupancy 属性上未设置 Occupied 位，则 UnoccupiedCoolingSetpoint 属性应设置为 CoolingSetpoint。
- b) 否则，OccupiedCoolingSetpoint 属性应设置为 CoolingSetpoint。

如果使用 HeatingSetpoint 来确定加热设定点：

- a) 如果服务端支持 OCC 功能，并且 Occupancy 属性上未设置 Occupied 位，则 UnoccupiedHeatingSetpoint 属性应设置为 HeatingSetpoint。
- b) 否则，OccupiedHeatingSetpoint 属性应设置为 HeatingSetpoint。如果以下所有情况都为真，则 ScheduleTransitionStruct 应无效：

- 1) 未提供 HeatingSetpoint 字段
- 2) 未提供 PresetHandle 字段
- 3) 未提供包含 ScheduleStruct 的 PresetHandle 字段
- 4) 提供了 SystemMode 字段，其值为 Heat 或 Auto，或者父 ScheduleStruct 上的 SystemMode 字段的值为 Heat 或 Auto

如果以下所有情况都为真，则 ScheduleTransitionStruct 应无效：

- 1) 未提供 CoolingSetpoint 字段
- 2) 未提供 PresetHandle 字段
- 3) 未提供包含 ScheduleStruct 的 PresetHandle 字段
- 4) 提供了 SystemMode 字段，其值为 Cool 或 Auto，或者父 ScheduleStruct 上的 SystemMode 字段的值为 Cool 或 Auto

——DayOfWeek 字段

此字段应指定转换适用的星期几的位掩码。Vacationbit 不应被设置；vacationschedules 应通过休假预设进行设置。

——TransitionTime 字段

这应指定转换生效的时间,以代表挂钟的一天内的分钟数表示,其中0表示00:00:00,1表示00:01:00,1439表示23:59:00。

夏令时转换期间的转换处理取决于具体实现。

——PresetHandle 字段

此字段应指定在TransitionTime使用的预设。如果提供此字段,则不应提供SystemMode、CoolingSetpoint和HeatingSetpoint字段。

——SystemMode 字段

此项应指定恒温器在转换时切换的默认模式,以覆盖计划的默认模式。此字段的唯一有效值应为自动、加热、冷却和关闭。仅当所需系统模式与计划的默认SystemMode不同时,才应包含此字段。

——CoolingSetpoint 字段

此字段应指定转换的冷却设定点。如果设置了PresetHandle,则不应包含此字段。请参阅“6.6.3.7 设定点限制”以了解限制。

——HeatingSetpoint 字段

此字段应指定转换的加热设定点。如果设置了PresetHandle,则不应包含此字段。请参阅“6.6.3.7 设定点限制”以了解限制。

### 6.5.3.8.33 ScheduleTypeStruct 类型

ScheduleTypeStruct 类型描述详见下表256。

表256 ScheduleTypeStruct 类型

| 标识符 | 名称                   | 类型                         | 约束条件                 | 质量等级 | 默认 | 通道 | 一致性 |
|-----|----------------------|----------------------------|----------------------|------|----|----|-----|
| 0   | SystemMode           | SystemModeEnum             | desc                 | —    | —  | —  | M   |
| 1   | NumberOfSchedules    | uint8                      | maxNumberOfSchedules | —    | 0  | —  | M   |
| 2   | ScheduleTypeFeatures | ScheduleTypeFeaturesBitmap | desc                 | —    | 0  | —  | M   |

其中,

- SystemMode字段应指定此恒温器支持的用于时间表的SystemModeEnum。此字段的唯一有效值应为自动、加热和冷却。
- NumberOfSchedules字段应指定此系统模式的计划数量的限制。
- ScheduleTypeFeatures字段应指定此计划条目的功能位图。应至少设置SupportsPresets和 SupportsSetpoints之一。

### 6.5.3.9 属性

恒温功能集属性见下表 257。

表257 恒温功能集属性表

| 标识符    | 名称                      | 类型                                 | 约束条件     | 质量等级 | 默认   | 通道 | 一致性    |
|--------|-------------------------|------------------------------------|----------|------|------|----|--------|
| 0x0000 | LocalTemperature        | temperature                        | all      | XP   | null | RV | M      |
| 0x0001 | OutdoorTemperature      | temperature                        | all      | X    | null | RV | 0      |
| 0x0002 | Occupancy               | 章节“6.6.3.9.7<br>OccupancyBitmap类型” | all      |      | 1    | RV | OCC    |
| 0x0003 | AbsMinHeatSetpointLimit | temperature                        | desc     | F    | 7°C  | RV | [HEAT] |
| 0x0004 | AbsMaxHeatSetpointLimit | temperature                        | desc     | F    | 30°C | RV | [HEAT] |
| 0x0005 | AbsMinCoolSetpointLimit | temperature                        | desc     | F    | 16°C | RV | [COOL] |
| 0x0006 | AbsMaxCoolSetpointLimit | temperature                        | desc     | F    | 32°C | RV | [COOL] |
| 0x0007 | PICoolingDemand         | uint8                              | 0%to100% | P    | —    | RV | [COOL] |

表257 恒温功能集属性表（续）

| 标识符    | 名称                                 | 类型                             | 约束条件                                   | 质量等级 | 默认                      | 通道     | 一致性      |
|--------|------------------------------------|--------------------------------|----------------------------------------|------|-------------------------|--------|----------|
| 0x0008 | PIHeatingDemand                    | uint8                          | 0%to100%                               | P    | -                       | RV     | [HEAT]   |
| 0x0009 | HVACSystemTypeConfiguration        | HVACSystemTypeBitmap           | desc                                   | N    | 0                       | R[W]VM | D        |
| 0x0010 | LocalTemperatureCalibration        | SignedTemperature              | all                                    | N    | 0°C                     | RWVM   | [!LTNE]  |
| 0x0011 | OccupiedCoolingSetpoint            | temperature                    | desc                                   | N    | 26°C                    | RWVO   | COOL     |
| 0x0012 | OccupiedHeatingSetpoint            | temperature                    | desc                                   | N    | 20°C                    | RWVO   | HEAT     |
| 0x0013 | UnoccupiedCoolingSetpoint          | temperature                    | desc                                   | N    | 26°C                    | RWVO   | COOL&OCC |
| 0x0014 | UnoccupiedHeatingSetpoint          | temperature                    | desc                                   | N    | 20°C                    | RWVO   | HEAT&OCC |
| 0x0015 | MinHeatSetpointLimit               | temperature                    | desc                                   | N    | AbsMinHeatSetpointLimit | RWVM   | [HEAT]   |
| 0x0016 | MaxHeatSetpointLimit               | temperature                    | desc                                   | N    | AbsMaxHeatSetpointLimit | RWVM   | [HEAT]   |
| 0x0017 | MinCoolSetpointLimit               | temperature                    | desc                                   | N    | AbsMinCoolSetpointLimit | RWVM   | [COOL]   |
| 0x0018 | MaxCoolSetpointLimit               | temperature                    | desc                                   | N    | AbsMaxCoolSetpointLimit | RWVM   | [COOL]   |
| 0x0019 | MinSetpointDeadBand                | SignedTemperature              | 0to12.7°C                              | N    | 2.5°C                   | R[W]VM | AUTO     |
| 0x001A | RemoteSensing                      | RemoteSensingBitmap            | 00000xxx                               | N    | 0                       | RWVM   | O        |
| 0x001B | ControlSequenceOfOperation         | ControlSequenceOfOperationEnum | desc                                   | N    | 4                       | RWVM   | M        |
| 0x001C | SystemMode                         | SystemModeEnum                 | desc                                   | N    | 1                       | RWVM   | M        |
| 0x001D | AlarmMask                          | AlarmCodeBitmap                | desc                                   | -    | 0                       | RV     | [Zigbee] |
| 0x001E | ThermostatRunningMode              | ThermostatRunningModeEnum      | desc                                   | -    | 0                       | RV     | [AUTO]   |
| 0x0020 | StartOfWeek                        | StartOfWeekEnum                | desc                                   | F    | -                       | RV     | SCH      |
| 0x0021 | NumberOfWeeklyTransitions          | uint8                          | all                                    | F    | 0                       | RV     | SCH      |
| 0x0022 | NumberOfDailyTransitions           | uint8                          | all                                    | F    | 0                       | RV     | SCH      |
| 0x0023 | TemperatureSetpointHold            | TemperatureSetpointHoldEnum    | desc                                   | N    | 0                       | RWVM   | O        |
| 0x0024 | TemperatureSetpointHoldDuration    | uint16                         | max1440                                | NX   | null                    | RWVM   | O        |
| 0x0025 | ThermostatProgrammingOperationMode | ProgrammingOperationModeBitmap | desc                                   | P    | 0                       | RWVM   | O        |
| 0x0029 | ThermostatRunningState             | RelayStateBitmap               | desc                                   | -    | -                       | RV     | O        |
| 0x0030 | SetpointChangeSource               | SetpointChangeSourceEnum       | desc                                   | -    | 0                       | RV     | O        |
| 0x0031 | SetpointChangeAmount               | TemperatureDifference          | all                                    | X    | null                    | RV     | O        |
| 0x0032 | SetpointChangeSourceTimestamp      | utc                            | all                                    |      | 0                       | RV     | O        |
| 0x0034 | OccupiedSetback                    | UnsignedTemperature            | OccupiedSetbackMintoOccupiedSetbackMax | XN   | null                    | RWVM   | SB       |

表257 恒温功能集属性表（续）

| 标识符    | 名称                               | 类型                       | 约束条件                                       | 质量等级 | 默认    | 通道   | 一致性    |
|--------|----------------------------------|--------------------------|--------------------------------------------|------|-------|------|--------|
| 0x0035 | OccupiedSetbackMin               | UnsignedTemperature      | maxOccupiedSetbackMax                      | XF   | null  | RV   | SB     |
| 0x0036 | OccupiedSetbackMax               | UnsignedTemperature      | OccupiedSetbackMinto25.4℃                  | XF   | null  | RV   | SB     |
| 0x0037 | UnoccupiedSetback                | UnsignedTemperature      | UnoccupiedSetbackMintoUnoccupiedSetbackMax | XN   | null  | RWVM | SB&OCC |
| 0x0038 | UnoccupiedSetbackMin             | UnsignedTemperature      | maxUnoccupiedSetbackMax                    | XF   | null  | RV   | SB&OCC |
| 0x0039 | UnoccupiedSetbackMax             | UnsignedTemperature      | UnoccupiedSetbackMinto25.4℃                | XF   | null  | RV   | SB&OCC |
| 0x003A | EmergencyHeatDelta               | UnsignedTemperature      | all                                        | N    | 25.5℃ | RWVM | 0      |
| 0x0040 | ACType                           | ACTypeEnum               | desc                                       | N    | 0     | RWVM | 0      |
| 0x0041 | ACCapacity                       | uint16                   | all                                        | N    | 0     | RWVM | 0      |
| 0x0042 | ACRefrigerantType                | ACRefrigerantTypeEnum    | desc                                       | N    | 0     | RWVM | 0      |
| 0x0043 | ACCompressorType                 | ACCompressorTypeEnum     | desc                                       | N    | 0     | RWVM | 0      |
| 0x0044 | ACErrorCode                      | ACErrorCodeBitmap        | all                                        | —    | 0     | RWVM | 0      |
| 0x0045 | ACLouverPosition                 | ACLouverPositionEnum     | desc                                       | N    | 0     | RWVM | 0      |
| 0x0046 | ACCoilTemperature                | temperature              | all                                        | X    | null  | RV   | 0      |
| 0x0047 | ACCapacityFormat                 | ACCapacityFormatEnum     | desc                                       | N    | 0     | RWVM | 0      |
| 0x0048 | PresetTypes                      | list[PresetTypeStruct]   | desc                                       | F    | MS    | RV   | PRES   |
| 0x0049 | ScheduleTypes                    | list[ScheduleTypeStruct] | desc                                       | F    | MS    | RV   | MSCH   |
| 0x004A | NumberOfPresets                  | uint8                    | all                                        | F    | 0     | RV   | PRES   |
| 0x004B | NumberOfSchedules                | uint8                    | all                                        | F    | 0     | RV   | MSCH   |
| 0x004C | NumberOfScheduleTransitions      | uint8                    | all                                        | F    | 0     | RV   | MSCH   |
| 0x004D | NumberOfScheduleTransitionPerDay | uint8                    | all                                        | XF   | null  | RV   | MSCH   |
| 0x004E | ActivePresetHandle               | octstr                   | max16                                      | XN   | null  | RV   | PRES   |
| 0x004F | ActiveScheduleHandle             | octstr                   | max16                                      | XN   | null  | RV   | MSCH   |
| 0x0050 | Presets                          | list[PresetStruct]       | maxNumberOfPresets                         | NT   | empty | RWVM | PRES   |
| 0x0051 | Schedules                        | list[ScheduleStruct]     | desc                                       | NT   | empty | RWVM | MSCH   |
| 0x0052 | SetpointHoldExpiryTimestamp      | epoch-s                  | all                                        | XN   | null  | RV   | 0      |

## ——计算本地温度

本地温度应根据measuredtemperature计算得出，包括根据LocalTemperatureCalibration属性（如果有）应用的任何调整，如下公式（11）所示：

$$\text{CalculatedLocalTemperature} = (\text{measuredtemperature}) + \text{LocalTemperatureCalibration} \quad (11)$$

式中：measuredtemperature 的值可能是本地的，也可能是远程的，这取决于 RemoteSensing 属性的值（如果支持的话）。

恒温器功能集中的所有设定点属性都应基于这个计算值（即测量的温度和任何校准偏移）来触发。如果存在LocalTemperatureNotExposed功能，则恒温器的行为应为：即使该值未在LocalTemperature属性中上报，设备的温度控制仍然使用计算后的本地温度（Calculated Local Temperature）。

——LocalTemperature 属性

此属性应该在可用时指示当前的CalculatedLocalTemperature。

- 如果不支持LTNE功能：
  - 如果LocalTemperatureCalibration无效或当前不可用，该属性应上报null。
  - 如果LocalTemperatureCalibration有效，该属性应上报该值。
- 反之，若支持LTNE功能，则LocalTemperatureCalibration不会有外部反馈。在这种情况下，

LocalTemperature属性应始终上报null。

——OutdoorTemperature 属性

此属性应指示测量到的本地或远程（通过网络）室外温度。

——Occupancy 属性

该属性应指示加热/冷却的空间内是否有人，该属性可通过本地或远程（通过网络）测量得到。

——AbsMinHeatSetpointLimit 属性

该属性应指示加热设定点可以被设定的绝对最小值。这是制造商规定的限制。

参见“6.6.3.7 设定点限制”以了解约束。

——AbsMaxHeatSetpointLimit 属性

该属性应指示加热设定点可以被设定的绝对最大值。这是制造商规定的限制。

参见“6.6.3.7 设定点限制”以了解约束。

——AbsMinCoolSetpointLimit 属性

该属性应指示冷却设定点可以被设定的绝对最小值。这是制造商规定的限制。

参见“6.6.3.7 设定点限制”以了解约束。

——AbsMaxCoolSetpointLimit 属性

该属性应指示冷却设定点可以被设定的绝对最大值。这是制造商规定的限制。

参见“6.6.3.7 设定点限制”以了解约束。

——PICoolingDemand 属性

该属性应指示恒温器（如果有）使用的PI（比例积分）控制回路所需的冷却级别，以百分比表示。当恒温器处于“关闭”或“加热”模式时，该值为0。

该属性会定期上报，可用于控制冷却设备。

——PIHeatingDemand 属性

该属性应指示PI控制回路所需的加热级别，以百分比表示。当恒温器处于“关闭”或“冷却”模式时，该值为0。

该属性会定期上报，可用于控制加热设备。

——HVACSystemTypeConfiguration 属性

该属性应指示由恒温器控制的HVAC系统类型。如果恒温器使用物理DIP开关来设置这些参数，则这些信息应只能从DIP开关读取。如果这些参数是通过软件设置的，则应具有读/写权限，以提供远程编程能力。

——LocalTemperatureCalibration 属性

该属性应指示恒温器服务端在使用、显示或报告测量温度（本地或远程）之前，对CalculatedLocalTemperature进行调整的偏移量。

该属性的目的是根据用户的偏好调整恒温器服务端的校准（例如，确保多个服务端在同一HVAC区域显示一致的温度值），或补偿温度传感器之间的差异。

如果恒温器客户端尝试写入一个不被支持的LocalTemperatureCalibration属性值（例如，超出恒温器服务端支持的范围），恒温器服务端应响应成功状态，并将LocalTemperatureCalibration的值设置为所达到的上限或下限。

注：在本簇规范修订至第8版之前，该属性的取值范围被限定在-2.5° C到2.5° C之间。

——OccupiedCoolingSetpoint 属性

该属性应指示当房间有人时的冷却模式设定点。

参见“6.6.3.7 设定点限制”以了解约束。

如果尝试将此属性设置为大于MaxCoolSetpointLimit或小于MinCoolSetpointLimit的值，系统应返回带有状态码CONSTRAINT\_ERROR的响应。

如果将此属性值设置为小于  $(\text{OccupiedHeatingSetpoint} + \text{MinSetpointDeadBand})$  的值，则  $\text{OccupiedHeatingSetpoint}$  的值应调整为  $(\text{OccupiedCoolingSetpoint} - \text{MinSetpointDeadBand})$ 。

如果房间的人体存在状态未知，该属性应作为冷却模式的设定点使用。

如果客户端更改了此属性的值，并且服务端支持PRES功能，同时服务端不支持OCC功能或Occupancy属性中的Occupied位已被设置，那么ActivePresetHandle属性的值应被设为null。

——OccupiedHeatingSetpoint 属性

该属性用于指示房间有人时的加热模式设定点。

参见“6.6.3.7 设定点限制”以了解约束。

如果尝试将此属性设置为大于MaxHeatSetpointLimit或小于MinHeatSetpointLimit的值，系统应当返回带有状态码CONSTRAINT\_ERROR的响应。

如果将此属性设置为大于  $(\text{OccupiedCoolingSetpoint} - \text{MinSetpointDeadBand})$  的值，则  $\text{OccupiedCoolingSetpoint}$  的值应调整为  $(\text{OccupiedHeatingSetpoint} + \text{MinSetpointDeadBand})$ 。

如果房间的人体存在状态未知，该属性应作为加热模式的设定点使用。

如果客户端更改了此属性的值，并且服务端支持PRES功能，同时服务端不支持OCC功能或Occupancy属性中的Occupied位已被设置，那么ActivePresetHandle属性的值应被设为null。

——UnoccupiedCoolingSetpoint 属性

该属性用于指示房间无人时的冷却模式设定点。

参见“6.6.3.7 设定点限制”以了解约束。

如果尝试将此属性设置为大于MaxCoolSetpointLimit或小于MinCoolSetpointLimit的值，系统应返回带有状态码CONSTRAINT\_ERROR的响应。

如果将此属性值设置为小于  $(\text{UnoccupiedHeatingSetpoint} + \text{MinSetpointDeadBand})$  的值，则  $\text{UnoccupiedHeatingSetpoint}$  的值应调整为  $(\text{UnoccupiedCoolingSetpoint} - \text{MinSetpointDeadBand})$ 。

如果房间的人体存在状态未知，不应使用该属性。

如果客户端更改了此属性的值，并且服务端支持PRES功能，同时服务端不支持OCC功能或Occupancy属性中的Occupied位已被设置，那么ActivePresetHandle属性的值应被设为null。

——UnoccupiedHeatingSetpoint 属性

该属性用于指示房间无人时的加热模式设定点。

参见“6.6.3.7 设定点限制”以了解约束。

如果尝试将此属性设置为大于MaxHeatSetpointLimit或小于MinHeatSetpointLimit的值，系统应返回带有状态码CONSTRAINT\_ERROR的响应。

如果将此属性设置为大于  $(\text{UnoccupiedCoolingSetpoint} - \text{MinSetpointDeadBand})$  的值，则  $\text{UnoccupiedCoolingSetpoint}$  的值应调整为  $(\text{UnoccupiedHeatingSetpoint} + \text{MinSetpointDeadBand})$ 。

如果房间的人体存在状态未知，不应使用该属性。

如果客户端更改了此属性的值，并且服务端支持PRES功能，同时服务端不支持OCC功能或Occupancy属性中的Occupied位已被设置，那么ActivePresetHandle属性的值应被设为null。

——MinHeatSetpointLimit 属性

该属性应指示加热设定点可以设置的最低水平。

该属性以及接下来的三个属性允许用户自定义比制造商规定的更严格的设定点。将用户（例如，在商业建筑中）限制在这样的设定点内有助于节约电力。

参见“6.6.3.7 设定点限制”以了解约束。

如果尝试将该属性设置为与设定点值冲突的值，则应通过最小调整设定点来允许该属性设置为所需值。如果尝试将该属性设置为与约束条件不一致且无法通过修改设定点解决的值，则应返回状态码为CONSTRAINT\_ERROR的响应。

——MaxHeatSetpointLimit 属性

该属性应指示加热设定点可以设置的最大水平。

参见“6.6.3.7 设定点限制”以了解约束。如果尝试将该属性设置为与设定点值冲突的值，则应通过最小调整设定点来允许将该属性设置为所需值。如果尝试将该属性设置为不符合约束条件且无法通过修改设定点来解决的值，则应返回状态码为CONSTRAINT\_ERROR的响应。

## ——MinCoolSetpointLimit 属性

该属性应指示冷却设定点可以设置的最低水平。

参见“6.6.3.7 设定点限制”以了解约束。如果尝试将该属性设置为与设定点值冲突的值，则应通过最小调整量来调整这些设定点，以允许将该属性设置为所需值。如果尝试将该属性设置为不符合约束条件且无法通过修改设定点来解决的值，则应返回状态码为CONSTRAINT\_ERROR的响应。

## ——MaxCoolSetpointLimit 属性

该属性应指示冷却设定点可以设置的最大水平。

参见“6.6.3.7 设定点限制”以了解约束。如果尝试将该属性设置为与设定点值冲突的值，则应通过最小调整设定点来允许将该属性设置为所需值。如果尝试将该属性设置为不符合约束条件且无法通过修改设定点解决的值，则应返回状态码为CONSTRAINT\_ERROR的响应。

## ——MinSetpointDeadBand 属性

在支持自动功能的设备上，此属性应表示加热设定点和冷却设定点之间的最小差异。

参见“6.6.3.7 设定点限制”以了解约束。

注：在本簇规范修订至第8版之前，该属性的取值范围被限定在0°C到2.5°C之间。为了向后兼容，此属性是可选的可写属性。然而，对该属性的任何写操作都应被默默忽略。

## ——RemoteSensing 属性

该属性应指示本地温度、室外温度和占用情况是否由远程网络传感器而非内部传感器感测。如果服务端中存在LTNE功能，则LocalTemperatureRemoteSensing位值应始终设置为0。

如果在存在LTNE功能时，将LocalTemperatureRemoteSensing位写入值1，则写入操作失败，服务端应报告CONSTRAINT\_ERROR。

## ——ControlSequenceOfOperation 属性

该属性应指示恒温器的整体操作环境，从而指示恒温器可以操作的可能系统模式。

如果尝试写入该属性，服务端应默默忽略该写入操作，并且该属性的值应保持不变。此行为是为了与现有恒温器的向后兼容性。

## ——SystemMode 属性

此属性应指示恒温器的当前运行模式。其值应受ControlSequenceOfOperation属性的限制。

## ——AlarmMask 属性

该属性应指示AlarmCodeBitmap中的每个警报是否已启用。

当设备上实现了警报功能集，并且发生了AlarmCodeBitmap中包含的某个警报条件时，将生成警报通知，并将警报代码字段设置为AlarmCodeBitmap中列出的值。

## ——ThermostatRunningMode 属性

该属性应指示恒温器的运行模式。此属性使用与SystemModeEnum相同的值，但只能为Off、Cool或Heat。此属性用于在恒温器的系统模式处于自动模式时提供附加信息。

## ——StartOfWeek 属性

该属性应指示该恒温器认为是每周设定点调度开始的星期几。通过读取该属性，可以将其用作确定设备是否支持每周调度的基础。成功响应意味着支持每周调度。

## ——NumberOfWeeklyTransitions 属性

该属性应指示恒温器能够处理的每周计划转换次数。

## ——NumberOfDailyTransitions 属性

此属性应指示恒温器能够处理多少个每日时间表转换。

## ——TemperatureSetpointHold 属性

该属性应指示恒温器的温度保持状态。如果保持状态为开启，恒温器应维持当前模式的温度设定点，直到系统模式发生变化。如果保持状态为关闭，恒温器应遵循其内部调度程序指定的设定点转换。如果恒温器支持特定持续时间的设定点保持，它还应实现TemperatureSetpointHoldDuration属性。

如果服务端支持特定持续时间的设定点保持，它还应实现SetpointHoldExpiryTimestamp属性。如果该属性更新为SetpointHoldOn，并且TemperatureSetpointHoldDuration具有非空值且支持SetpointHoldExpiryTimestamp，服务端应将SetpointHoldExpiryTimestamp更新为当前UTC时间戳（以秒为单位）加上TemperatureSetpointHoldDuration的值乘以60。如果该属性更新为SetpointHoldOff且支持SetpointHoldExpiryTimestamp，服务端应将SetpointHoldExpiryTimestamp设置为空。

## ——TemperatureSetpointHoldDuration 属性

该属性应指示设定点保持活动的时间，以分钟为单位。支持指定持续时间保持的恒温器应实现此属性。空值表示该字段未使用。所有其他值均为保留值。

如果此属性更新为非空值，并且 TemperatureSetpointHold 设置为 SetpointHoldOn 且支持 SetpointHoldExpiryTimestamp，则服务端应将 SetpointHoldExpiryTimestamp 更新为当前UTC时间戳（以秒为单位）加上此属性的新值乘以60。

如果此属性设置为空值且支持 SetpointHoldExpiryTimestamp，则服务端应将 SetpointHoldExpiryTimestamp 设置为空

#### ——ThermostatProgrammingOperationMode 属性

该属性应指示恒温器编程的操作状态。当客户端修改该属性时，恒温器应修改其编程操作，并在用户本地修改其编程操作时更新该属性。恒温器可以支持多个活动的 ProgrammingOperationModeBitmap。例如，恒温器可以同时计划在计划编程模式和恢复模式下运行。

包含计划的恒温器可以使用该属性来控制如何使用该计划，即使它们不支持计划配置功能。

当 ScheduleActive 未设置时，设定点仅通过恒温器上的手动上下调整或远程调整来改变，而不是通过内部计划编程来改变。

注：修改 ScheduleActive 位不会清除或删除以前的每周计划编程配置。

#### ——ThermostatRunningState 属性

该属性应指示加热、冷却和风扇继电器的当前状态。未实现的输出应视为关闭状态。

#### ——SetpointChangeSource 属性

该属性应指示当前活动的 OccupiedCoolingSetpoint 或 OccupiedHeatingSetpoint 的来源（即，谁或什么决定了当前的设定点）。该属性使服务提供商能够确定设定点的变化是由于居住者的舒适度、预定的编程还是其他来源（例如，电力公司或其他服务提供商）引起的。由于自动化服务可能会频繁发起设定点的变化，该属性可以清楚地区分在恒温器上进行的设定点变化的来源。

#### ——SetpointChangeAmount 属性

该属性应指示当前活动的 OccupiedCoolingSetpoint 或 OccupiedHeatingSetpoint 与之前活动设定点之间的差值。该属性旨在配合 SetpointChangeSource 属性使用；实现 SetpointChangeAmount 的设备也应实现 SetpointChangeSource。

null 值表示之前的设定点未知。

#### ——SetpointChangeSourceTimestamp 属性

此属性应指示记录 SetpointChangeAmount 属性更改时的 UTC 时间。

#### ——OccupiedSetback 属性

该属性应指示恒温器服务端允许计算的本地温度在启动状态变化以将温度恢复到用户期望的设定点之前，浮动在 OccupiedCoolingSetpoint（即，OccupiedCoolingSetpoint+OccupiedSetback）之上或 OccupiedHeatingSetpoint（即，OccupiedHeatingSetpoint–OccupiedSetback）之下的量。该属性有时也称为“跨度”。

该属性的目的是允许远程配置期望设定点和测量温度之间的跨度，以帮助防止过度循环并减少能源账单，尽管这可能会导致某些用户的舒适度降低。null 值表示该属性未使用。

如果恒温器客户端尝试将 OccupiedSetback 写入大于 OccupiedSetbackMax 的值，恒温器服务端应将其 OccupiedSetback 值设置为 OccupiedSetbackMax，并应发送带有状态代码字段枚举为 SUCCESS 响应的写属性响应命令。

如果恒温器客户端尝试将 OccupiedSetback 写入小于 OccupiedSetbackMin 的值，恒温器服务端应将其 OccupiedSetback 值设置为 OccupiedSetbackMin，并应发送带有状态代码字段枚举为 SUCCESS 响应的写属性响应命令。

#### ——OccupiedSetbackMin 属性

该属性应指示用户可以将 OccupiedSetback 属性配置到的最小值。空值表示该属性未被使用。

#### ——OccupiedSetbackMax 属性

该属性应指示用户可以配置的 OccupiedSetback 属性的最大值。空值表示该属性未被使用。

#### ——UnoccupiedSetback 属性

该属性应指示恒温器服务端允许计算的本地温度在 UnoccupiedCoolingSetpoint（即 UnoccupiedCoolingSetpoint+UnoccupiedSetback）之上或 UnoccupiedHeatingSetpoint（即

UnoccupiedHeatingSetpoint-UnoccupiedSetback) 之下浮动的量, 然后再启动状态变化以将温度恢复到用户的期望设定点。该属性有时也称为“跨度”。

该属性的目的是允许远程配置期望设定点和测量温度之间的跨度, 以帮助防止过度循环并降低能源费用, 尽管这可能会导致某些用户的舒适度降低。

null值表示该属性未被使用。

如果恒温器客户端尝试将UnoccupiedSetback写入一个大于UnoccupiedSetbackMax的值, 恒温器服务端应将其UnoccupiedSetback设置为UnoccupiedSetbackMax, 并应发送一个带有状态代码字段枚举为成功响应的写属性响应命令。

如果恒温器客户端尝试将UnoccupiedSetback写入一个应小于UnoccupiedSetbackMin, 恒温器服务端应将其UnoccupiedSetback设置为UnoccupiedSetbackMin, 并应发送一个带有状态代码字段枚举为成功响应的写属性响应命令。

——UnoccupiedSetbackMin 属性

该属性应指示用户配置UnoccupiedSetback属性时, 恒温器服务端允许的最小值。

null值表示该属性未被使用。

——UnoccupiedSetbackMax 属性

该属性应指示用户可以配置的恒温器服务端允许的UnoccupiedSetback属性的最大值。

null值表示该属性未被使用。

——EmergencyHeatDelta 属性

该属性应指示计算的本地温度与OccupiedHeatingSetpoint或UnoccupiedHeatingSetpoint属性之间的差值, 恒温器服务端将在此差值下进入紧急加热模式。

如果计算的本地温度与OccupiedCoolingSetpoint或UnoccupiedCoolingSetpoint之间的差值大于或等于EmergencyHeatDelta, 并且恒温器服务端的SystemMode属性处于与加热相关的模式, 则恒温器服务端应立即切换到提供最高加热阶段(例如, 紧急加热)的系统模式属性值, 并继续在该运行状态下运行, 直到达到OccupiedHeatingSetpoint。例如:

- CalculatedLocalTemperature=10.0°C
- OccupiedHeatingSetpoint=16.0°C
- EmergencyHeatDelta=2.0°C
- OccupiedHeatingSetpoint-CalculatedLocalTemperature $\geq$ ?EmergencyHeatDelta
- 16°C-10°C $\geq$ ?2°C
- TRUE>>>恒温器服务端将其SystemMode更改为在2nd阶段或紧急加热模式下运行

此属性的目的是为恒温器客户端提供快速加热的能力当设定点比测量温度高出指定量时。这允许加热空间快速加热到用户设置的所需水平。

——ACType 属性

此属性应指示MiniSplitACTypeEnum的类型, MiniSplitAC的类型根据MiniSplitAC如何实现冷却和加热条件而定义。

——ACCapacity 属性

此属性应指示以ACCapacityFormat属性定义的格式表示的MiniSplitAC的容量。

——ACRefrigerantType 属性

此属性应指示MiniSplitAC内部使用的冷却剂类型。

——ACCompressorType 属性

此属性应指示MiniSplitAC内部使用的压缩机类型。

——ACErrorCode 属性

此属性应指示MiniSplitAC内部遇到的错误类型。

——ACLouverPosition 属性

此属性应指示百叶窗在空调上的位置。

——ACCoilTemperature 属性

此属性应指示空调线圈的温度, 可以是本地测量的, 也可以是远程(通过网络)测量的。

——ACCapacityFormat 属性

此属性应指示ACCapacity的格式。

——PresetTypes 属性

此属性应指示支持的PresetScenarioEnum值、每个PresetScenarioEnum可以创建的预设数量限制，以及恒温器是否可以自动切换到给定的场景。

——ScheduleTypes 属性

此属性应指示支持的系统模式SystemMode值、每个系统模式值可以创建的计划数量限制，以及给定的系统模式值是否支持转换到预设值、目标设定点或两者兼有。

——NumberOfPresets 属性

此属性应指示预设属性支持的最大entries数。

——NumberOfSchedules 属性

此属性应指示Schedules属性支持的最大entries数。

——NumberOfScheduleTransitions 属性

此属性应指示每个Schedules属性条目中允许的最大transitions数量。

——NumberOfScheduleTransitionsPerDay 属性

此属性应指示每个Schedules属性条目每天支持的最大transitions数量。如果此值为空，则每天的transitions数量没有限制。

——ActivePresetHandle 属性

此属性应指示当前激活预设的PresetHandle。如果此属性为null，则表示没有激活的预设。

——ActiveScheduleHandle 属性

此属性应指示当前激活时间表的ScheduleHandle。如果此属性为null，则表示没有激活的时间表。

——Presets 属性

此属性应包含当前配置的presets列表。在收到写入请求时：

- a) 如果PresetHandle字段为空，则PresetStruct应视为新增的预设，设备应为PresetHandle字段创建一个新的唯一值。
  - 1) 如果BuiltIn字段为true，应返回状态码CONSTRAINT\_ERROR的响应。
- b) 如果PresetHandle字段不为空，则PresetStruct应视为现有预设的修改。
  - 1) 如果PresetHandle字段的值与任何现有的presets不匹配，应返回状态码NOT\_FOUND的响应。
  - 2) 如果更新列表中的多个presets的PresetHandle字段值重复，应返回状态码CONSTRAINT\_ERROR的响应。
  - 3) 如果BuiltIn字段为true，而当前值中匹配PresetHandle字段的PresetStruct的BuiltIn字段为false，应返回状态码CONSTRAINT\_ERROR的响应。
  - 4) 如果BuiltIn字段为false，而当前值中匹配PresetHandle字段的PresetStruct的BuiltIn字段为true，应返回状态码CONSTRAINT\_ERROR的响应。
- c) 如果指定的PresetScenarioEnum值在PresetTypes中不存在，应返回状态码CONSTRAINT\_ERROR的响应。
- d) 如果设置了Name，但关联的PresetTypeStruct没有设置SupportsNames位，应返回状态码CONSTRAINT\_ERROR的响应。
- e) 如果将接收到的PresetStruct添加到待处理的Presets列表会导致待处理presets的总数超过NumberOfPresets属性的值，应返回状态码RESOURCE\_EXHAUSTED的响应。
- f) 如果将接收到的PresetStruct添加到待处理的Presets列表，会导致待处理预设中PresetScenario字段与已添加预设的PresetScenario字段匹配的presets的总数超过PresetTypeStruct中与该PresetScenario匹配的NumberOfPresets字段的值，此时应返回状态码RESOURCE\_EXHAUSTED的响应。
- g) 否则，写入请求应处于待处理状态，直到收到提交请求，并返回状态码SUCCESS的响应。
 

如果BuiltIn字段为null：

  - 1) 如果当前值中存在一个PresetStruct，其PresetHandle字段与接收的PresetStruct匹配，则待处理PresetStruct的BuiltIn字段应设置为匹配PresetStruct的BuiltIn值。
  - 2) 否则，待处理PresetStruct的BuiltIn字段应设置为false。

在尝试提交时，此属性的状态应按以下方式确定：

- a) 对于所有现有的presets：

如果在应用所有待处理更改后，更新的Presets属性值中不包含与PresetHandle字段匹配的PresetStruct，表明该PresetStruct被移除，则服务端应检查是否非法移除该PresetStruct：

- 1) 如果被移除的PresetStruct的BuiltIn字段为true，则属性状态应为CONSTRAINT\_ERROR。
- 2) 如果支持MSCH功能，并且被移除的PresetHandle在更新后的Schedules属性中被任何ScheduleStruct的任何ScheduleTransitionStruct中的PresetHandle引用，则属性状态应为INVALID\_IN\_STATE。
- 3) 如果被移除的PresetHandle等于ActivePresetHandle属性的值，则属性状态应为INVALID\_IN\_STATE。
- b) 否则，属性状态应为SUCCESS。

——Schedules属性

此属性应包含一个ScheduleStructs列表。

在接收到写入请求时：

- a) 对于写入请求中的所有schedules：
  - 1) 如果ScheduleHandle字段为null，ScheduleStruct应被视为新增的schedule，设备应为ScheduleHandle字段创建一个新的唯一值。
 

如果BuiltIn字段为true，应返回状态码CONSTRAINT\_ERROR。
  - 2) 否则，如果ScheduleHandle字段不为空，ScheduleStruct应被视为现有schedule的修改。
    - i. 如果ScheduleHandle字段的值与任何现有schedules不匹配，应返回状态码NOT\_FOUND。
    - ii. 如果BuiltIn字段为true，且当前值中与ScheduleHandle字段匹配的ScheduleStruct的BuiltIn字段设置为false，应返回状态码CONSTRAINT\_ERROR。
    - iii. 如果BuiltIn字段为false，且当前值中与ScheduleHandle字段匹配的ScheduleStruct的BuiltIn字段设置为true，应返回状态码CONSTRAINT\_ERROR。
  - 3) 如果指定的SystemMode不存在于ScheduleTypes中，应返回状态码CONSTRAINT\_ERROR。
  - 4) 如果transitions的数量超过NumberOfScheduleTransitions的值，应返回状态码RESOURCE\_EXHAUSTED。
  - 5) 如果NumberOfScheduleTransitionsPerDay属性的值不为空，且在某一天中的transitions数量超过NumberOfScheduleTransitionsPerDay的值，应返回状态码RESOURCE\_EXHAUSTED。
  - 6) 如果PresetHandle字段存在，但关联的ScheduleTypeStruct没有设置SupportsPresets位，应返回状态码CONSTRAINT\_ERROR。
  - 7) 如果PresetHandle字段存在，但在应用所有待处理更改后，Presets属性中不包含PresetHandle字段与PresetHandle值匹配的PresetStruct，应返回状态码CONSTRAINT\_ERROR。
  - 8) 如果Name被设置，但关联的ScheduleTypeStruct没有设置SupportsNames位，应返回状态码CONSTRAINT\_ERROR。
  - 9) 对于写入请求中所有schedules的所有transitions：
 

如果PresetHandle字段存在，但与ScheduleStruct的SystemMode字段匹配的ScheduleTypeStruct没有设置SupportsPresets位，应返回状态码CONSTRAINT\_ERROR。
  - 10) 如果PresetHandle字段存在，但在应用所有待处理更改后，Presets属性中不包含PresetHandle字段与PresetHandle值匹配的PresetStruct，应返回状态码CONSTRAINT\_ERROR。
    - i. 如果SystemMode字段存在，但与ScheduleStruct的SystemMode字段值匹配的ScheduleTypeStruct没有设置SupportsSetpoints位，应返回状态码CONSTRAINT\_ERROR。
    - ii. 如果SystemMode字段的值为SystemModeOff，但与ScheduleStruct的SystemMode字段值匹配的ScheduleTypeStruct没有设置SupportsOff位，应返回状态码CONSTRAINT\_ERROR。
  - 11) 如果HeatingSetpoint字段存在，但与ScheduleStruct的SystemMode字段值匹配的ScheduleTypeStruct没有设置SupportsSetpoints位，应返回状态码CONSTRAINT\_ERROR。
 

如果CoolingSetpoint字段存在，但与ScheduleStruct的SystemMode字段值匹配的ScheduleTypeStruct没有设置SupportsSetpoints位，应返回状态码CONSTRAINT\_ERROR。

- b) 如果将接收到的 ScheduleStruct 添加到 Schedules 的待处理列表中, 会导致待处理的 schedules 总数超过 NumberOfSchedules 属性的值, 此时应返回状态码 RESOURCE\_EXHAUSTED。
- c) 如果将接收到的 ScheduleStruct 添加到 Schedules 的待处理列表中, 会导致待处理的 schedules 中 SystemMode 字段匹配新增的 schedule 的 SystemMode 字段的 schedule 总数超过与新增 schedule 的 SystemMode 字段匹配的 ScheduleTypeStruct 的 NumberOfSchedules 值, 此时应返回状态码 RESOURCE\_EXHAUSTED。
- d) 否则, 写入应挂起, 直到接收到提交请求, 并返回状态码 SUCCESS。

如果BuiltIn字段为null:

- 1) 如果当前值中存在 ScheduleHandle 字段匹配的 ScheduleStruct, 待处理的 ScheduleStruct 的 BuiltIn 字段应设置为匹配的 ScheduleStruct 的 BuiltIn 的值。
- 2) 否则, 待处理的 ScheduleStruct 的 BuiltIn 字段应设置为 false。

尝试提交时, 此属性的状态应按以下方式确定:

- a) 对于所有现有的 schedules:
  - 如果在应用所有待处理更改后, Schedules属性的更新值中不包含ScheduleHandle字段匹配的 ScheduleStruct, 表明ScheduleStruct被移除, 服务端应检查是否非法移除ScheduleStruct:
  - 1) 如果被移除的 ScheduleStruct 的 BuiltIn 字段为 true, 则属性状态应为 CONSTRAINT\_ERROR。
  - 2) 如果被移除的 ScheduleHandle 等于 ActiveScheduleHandle 属性的值, 则属性状态应为 INVALID\_IN\_STATE。
- b) 否则, 属性状态应为 SUCCESS。

——SetpointHoldExpiryTimestamp 属性

如果有已知的时间, 当TemperatureSetpointHold应该被清除时, 此属性应包含该时间的UTC时间戳, 指示何时会发生清除。如果没有此已知时间, 则此属性应为null。

如果TemperatureSetpointHold被设置为SetpointHoldOff或者TemperatureSetpointHoldDuration 设置为null, 则此属性应设置为null, 表示在恒温器上没有保持状态, 无论是否设置持续时间。

### 6.5.3.10 命令

#### 6.5.3.10.1 命令列表

恒温功能集命令见下表258。

表258 恒温功能集命令表

| 标识符  | 名称                        | 方向                | 返回                        | 通道 | 一致性               |
|------|---------------------------|-------------------|---------------------------|----|-------------------|
| 0x00 | Set-pointRaiseLower       | client→<br>server | Y                         | 0  | M                 |
| 0x01 | SetWeeklySchedule         | client→<br>server | Y                         | M  | SCH               |
| 0x02 | GetWeeklySchedule         | client→<br>server | GetWeeklyScheduleResponse | 0  | SCH               |
| 0x00 | GetWeeklyScheduleResponse | client←<br>server | N                         | —  | SCH               |
| 0x03 | ClearWeeklySchedule       | client→<br>server | Y                         | M  | SCH               |
| 0x04 | GetRelayStatusLog         | client→<br>server | GetRelayStatusLogResponse | 0  | [Zigbee]          |
| 0x01 | GetRelayStatusLogResponse | client←<br>server | N                         | —  | GetRelayStatusLog |
| 0x05 | SetActiveScheduleRequest  | client→<br>server | Y                         | 0  | MSCH              |
| 0x06 | SetActivePresetRequest    | client→<br>server | Y                         | 0  | PRES              |

#### 6.5.3.10.2 SetpointRaiseLower 命令

SetpointRaiseLower 命令包含以下数据字段，详见下表259。

表259 SetpointRaiseLower 命令数据字段

| 标识符 | 名称     | 类型                         | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|--------|----------------------------|------|------|----|-----|
| 0   | Mode   | SetpointRaiseLowerModeEnum | desc | —    | —  | M   |
| 1   | Amount | int8                       | all  | —    | —  | M   |

其中，

——Mode 字段

此字段应指定哪些设定点需要被调整。

a) Heat 值

如果服务端不支持HEAT功能，则应响应INVALID\_COMMAND。如果服务端支持AUTO功能，并且由于MinSetpointDeadBand导致结果设定点无效，则应适当提高Cooling设定点，以维持死区（deadband）。

b) Cool 值

如果服务端不支持COOL功能，则应响应INVALID\_COMMAND。如果服务端支持AUTO功能，并且由于MinSetpointDeadBand导致结果设定点无效，则应适当降低Heating设定点，以维持死区（deadband）。

c) Both 值

客户端可以指示Both，而不考虑服务端的功能支持。服务端应只调整它支持的设定点，并且不响应错误。

——Amount 字段

此字段应指示应添加到设定点的数量（可能为负数），以 0.1℃为步长。

——EffectonReceipt

接收后，所指示的设定点的属性应添加Amount字段中指定的数量。如果结果值超出了MinCoolSetpointLimit、MaxCoolSetpointLimit、MinHeatSetpointLimit和MaxHeatSetpointLimit所施加的限制，则该值将被限制在这些限制内。这不会被视作错误情况。

6.5.3.10.3 SetWeeklySchedule 命令

该命令用于通过管理系统更新恒温器的每周设定点schedule。如果恒温器已经设置了每周设定点schedule，那么在接收到来自管理系统的更新时，它应该替换每一天的设定点。例如，如果恒温器原本为每周每一天都设定了4个设定点，而系统发送了一个包含星期六设定点的SetWeeklySchedule命令，那么恒温器应当删除星期六的所有4个设定点，并用新的设定点替换它们，同时其他日子的设定点不做更改。如果schedule的数据量超过单个数据帧的容量或包含超过10个transitions，那么该schedule将通过多个SetWeeklySchedule命令分批发送。详见下表260。

表260 SetWeeklySchedule 命令数据字段

| 标识符 | 名称                             | 类型                                   | 约束条件  | 质量等级 | 默认 | 一致性 |
|-----|--------------------------------|--------------------------------------|-------|------|----|-----|
| 0   | NumberOfTransitionsForSequence | uint8                                | all   | —    | —  | M   |
| 1   | DayOfWeekForSequence           | ScheduleDayOfWeekBitmap              | desc  | —    | —  | M   |
| 2   | ModeForSequence                | ScheduleModeBitmap                   | desc  | —    | —  | M   |
| 3   | Transitions                    | list[WeeklyScheduleTransitionStruct] | max10 | —    | —  | M   |

其中，

——NumberOfTransitionsForSequence 字段

该字段表示此命令序列中预计会有多少个独立的transitions。如果设备的schedule中支持超过10个transitions，可以通过发送多个“SetWeeklySchedule”命令来传递这些信息，每个命令包含设备设置所需的独立信息。

——DayOfWeekForSequence 字段

该字段表示命令负载中所有transitions所关联的星期几。这个字段是一个位图，因此关联的设定点可能跨越多个日期（例如，你可以为所有“工作日”设置一个变化时间，或者根据实现需求设置不同日期组合的变化）。每个设定点变化将从该变化的星期几开始。每个命令最多可以包含10个transitions。

——ModeForSequence 字段

该字段指示应用程序如何解码Transitions列表中的每个transition中的设定点字段。

如果HeatSetpointPresent位为On，则Transitions列表中的每一项的HeatSetpoint字段不得为空。

如果HeatSetpointPresent位为Off，则Transitions列表中的每一项的HeatSetpoint字段必须为空。

如果CoolSetpointPresent位为On，则Transitions列表中的每一项的CoolSetpoint字段不得为空。

如果CoolSetpointPresent位为Off，则Transitions列表中的每一项的CoolSetpoint字段必须为空。

ModeForSequence字节中的至少一个位必须为On。

即使设备不支持HEAT或COOL功能，也必须遵守这两个位的规则，以确保命令能够正确解码并处理。

——Transitions 字段

该字段包含用于更新指定每日schedule的设定点transitions列表。

——EffectonReceipt

接收命令后，每周更新设定点的schedule应该存储在恒温器中，并且应该从接收到命令的时间开始生效。系统应返回一个状态码作为响应。

当接收的命令所需的transitions总数超过设备支持的数量时，响应的状态码应为INSUFFICIENT\_SPACE。

当命令中发送的任何设定点超出范围（超出最小/最大设定点限制AbsMin/MaxSetPointLimit），或当ModeforSequence字段包含设备不支持的模式时，响应的状态码应为CONSTRAINT\_ERROR，且整个序列中的设定点不应被使用。

当检测到重叠的transition时，响应的状态码应为FAILURE。

当一个不支持多天命令的设备接收到DayOfWeekForSequence字段中设置了多个位的命令，或者当一个不支持多模式命令的设备接收到ModeForSequence字段中设置了多个位的命令，或者当Transitions字段的内容与NumberOfTransitionsForSequence、DayOfWeekForSequence或ModeForSequence字段不一致时，响应的状态码应为INVALID\_COMMAND。

当transitions成功添加时，响应的状态码应为SUCCESS。

6.5.3.10.4 GetWeeklySchedule 命令

GetWeeklySchedule 命令包含以下数据字段，详见下表261。

表261 GetWeeklySchedule 命令数据字段

| 标识符 | 名称            | 类型                      | 约束条件 | 质量等级 | 默认 | 一致性 |
|-----|---------------|-------------------------|------|------|----|-----|
| 0   | DaysToRe-turn | ScheduleDayOfWeekBitmap | desc | —    | —  | M   |
| 1   | ModeToRe-turn | ScheduleModeBitmap      | desc | —    | —  | M   |

——DaysToReturn 字段

该字段表示客户端希望返回多少天的设定点值，可以是单独某几天，也可以是整个一周的组合。

——ModeToReturn 字段

该字段表示客户端希望返回哪些模式的设定点值，可以是单独的加热模式、冷却模式，或者同时包括加热与冷却模式。

——EffectonReceipt

收到命令后，设备应返回GetWeeklyScheduleResponse命令。DaystoReturn和ModetoReturn字段使用位掩码（bitmask）定义，以支持在一个命令中灵活地处理多个日期和多个模式。如果恒温器无法处理包含多个日期和/或多个模式的命令，它应返回默认响应INVALID\_COMMAND。

6.5.3.10.5 GetWeeklyScheduleResponse 命令

该命令的负载格式与SetWeeklySchedule命令相同，详见下表262。

表262 GetWeeklyScheduleResponse 命令数据字段

| 标识符 | 名称                             | 类型                                   | 约束条件  | 质量等级 | 默认 | 一致性 |
|-----|--------------------------------|--------------------------------------|-------|------|----|-----|
| 0   | NumberOfTransitionsForSequence | uint8                                | all   | –    | –  | M   |
| 1   | DayOfWeekForSequence           | ScheduleDayOfWeekBitmap              | desc  | –    | –  | M   |
| 2   | ModeForSequence                | ScheduleModeBitmap                   | desc  | –    | –  | M   |
| 3   | Transitions                    | list[WeeklyScheduleTransitionStruct] | max10 | –    | –  | M   |

6.5.3.10.6 ClearWeeklySchedule 命令

该命令用于清除每周温度schedule。ClearWeeklySchedule命令没有负载。收到命令后，当前存储的所有温度变化点（transitions）应被清除，并且应返回默认的SUCCESS响应。该命令没有错误响应。

6.5.3.10.7 GetRelayStatusLog 命令

该命令用于查询恒温器内部的继电器状态日志。该命令没有负载。收到命令后，如果设备支持继电器状态日志功能，设备应以RelayStatusLog命令作为响应。日志的存储顺序为先进先出（FIFO），即日志生成并存储到队列中时，最早的记录会首先被替换。如果日志空间已满，新的记录将覆盖最旧的记录。日志的检索顺序为后进先出（LIFO），即客户端设备从队列中检索日志时，最新的记录会先被返回。一旦服务端发送了“GetRelayStatusLogResponse”帧，“UnreadEntries”属性应减少，表示队列中剩余的未读记录数量。

如果“UnreadEntries”属性为零，且客户端发送新的“GetRelayStatusLogRequest”请求，服务端可以返回以下其中一项作为响应：

- a) 重新发送最后一次的 GetRelayStatusLogResponse
  - 或者
  - b) 在请求时生成新的日志记录，并发送包含新数据的 GetRelayStatusLogResponse
- 在这两种情况下，“UnreadEntries”属性将保持为零。

6.5.3.10.8 GetRelayStatusLogResponse 命令

该命令由恒温器功能集服务端在响应GetRelayStatusLog请求时发送，详见下表。当继电器状态条目（RelayStatusEntry）通过无线传输发送到请求的客户端后，特定的条目将从恒温器的内部日志中清除。详见下表263。

表263 GetRelayStatusLogResponse 命令

| 标识符 | 名称                   | 类型               | 约束条件     | 质量等级 | 默认 | 一致性 |
|-----|----------------------|------------------|----------|------|----|-----|
| 0   | TimeOfDay            | uint16           | max1439  | –    | –  | M   |
| 1   | RelayStatus          | RelayStateBitmap | desc     | –    | –  | M   |
| 2   | LocalTemperature     | temperature      | all      | –    | –  | M   |
| 3   | HumidityInPercentage | uint8            | 0%to100% | X    | –  | M   |
| 4   | SetPoint             | temperature      | all      | –    | –  | M   |
| 5   | UnreadEntries        | uint16           | all      | –    | –  | M   |

其中，

- TimeOfDay字段表示继电器状态被记录时的时间，单位为自午夜以来的分钟数。例如，早上6点表示为360分钟，晚上11:30则表示为1410分钟。
- RelayStatus字段表示记录时恒温器的继电器状态。每一位代表恒温器使用的一个继电器。如果该位为1，表示关联的继电器处于开启和激活状态。每个恒温器制造商可以根据自己的设计定义位图与继电器之间的映射关系。
- LocalTemperature 字段表示记录时的 LocalTemperature。如果值为 null，表示 LocalTemperature无效或不可用。
- Humidity字段表示记录时的湿度，单位为百分比。如果值为null，表示湿度值无效或未知。
- Setpoint字段表示记录时的目标设定点温度。

- UnreadEntries字段表示恒温器内部日志系统中未读条目的数量。

6.5.3.10.9 SetActiveScheduleRequest 命令

SetActiveScheduleRequest 命令包含以下数据字段，详见下表264。

表264 SetActiveScheduleRequest 命令表

| 标识符 | 名称             | 类型     | 约束条件  | 质量等级 | 默认 | 一致性 |
|-----|----------------|--------|-------|------|----|-----|
| 0   | ScheduleHandle | octstr | max16 | –    | –  | M   |

——ScheduleHandle 字段

该字段指定要激活的ScheduleStruct中ScheduleHandle字段的值。

——EffectonReceipt

收到命令后，如果Schedules属性中包含一个ScheduleStruct，其ScheduleHandle字段与命令中ScheduleHandle字段的值匹配，服务端应将恒温器的ActiveScheduleHandle属性设置为该ScheduleHandle字段的值。否则，服务端应返回状态码INVALID\_COMMAND。

6.5.3.10.10 SetActivePresetRequest 命令

SetActivePresetRequest 命令包含以下数据字段，详见下表265。

表265 SetActivePresetRequest 命令表

| 标识符 | 名称           | 类型     | 约束条件  | 质量等级 | 默认 | 一致性 |
|-----|--------------|--------|-------|------|----|-----|
| 0   | PresetHandle | octstr | max16 | X    | –  | M   |

——PresetHandle 字段

该字段指定要激活的PresetStruct中PresetHandle字段的值。如果该字段设置为null，表示不应激活任何预设。

——EffectonReceipt

收到该命令后：

- a) 如果PresetHandle字段为null，服务端应通过将ActivePresetHandle属性设置为null来清除当前激活的预设。
- b) 否则：
  - 1) 如果收到的命令中PresetHandle对应的PresetHandle字段在Presets属性中没有匹配的PresetStruct，服务端应返回状态码INVALID\_COMMAND。
  - 2) 服务端应将ActivePresetHandle属性设置为PresetHandle字段的值。
- c) 服务端应返回状态码SUCCESS。

6.5.4 风扇控制功能集

6.5.4.1 概述

该功能集定义了一个接口，用于控制风扇的转速。

6.5.4.2 分类

风扇控制功能集分类详见下表266。

表266 风扇控制功能集分类表

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | FAN     |

6.5.4.3 Cluster ID

风扇控制功能集ID详见下表267。

表267 风扇控制功能集 ID 表

| 标识符    | 名称         |
|--------|------------|
| 0x0202 | FanControl |

#### 6.5.4.4 特性

本功能集应支持以下定义的Feature位图功能属性，详见下表268。

表268 风扇控制功能集特性表

| Bit | 编码   | 特征               | 概述          |
|-----|------|------------------|-------------|
| 0   | SPD  | MultiSpeed       | 0-最大风扇速度    |
| 1   | AUT  | Auto             | 支持自动模式的风扇速度 |
| 2   | RCK  | Rocking          | 支持摇摆动作      |
| 3   | WND  | Wind             | 支持模拟自然风     |
| 4   | STEP | Step             | 支持步进命令      |
| 5   | DIR  | AirflowDirection | 支持气流方向属性    |

##### ——MultiSpeed 特性

旧版的风扇控制Cluster版本0-1定义了3种速度（低速、中速和高速）以及自动速度控制，但它留给实现者来决定支持哪些功能。因此，可以假设传统的客户端实现能够从服务端获取支持的风速数量（1、2或3）以及是否支持自动速度控制。

MultiSpeed特性包括新的属性，支持从0到SpeedMax的风扇运行速度，最大值为100。

#### 6.5.4.5 数据类型

##### 6.5.4.5.1 RockBitmap 类型

RockBitmap 类型包含以下数据字段，详见下表269。

表269 RockBitmap 类型表

| Bit | 名称            | 概述     | 一致性 |
|-----|---------------|--------|-----|
| 0   | RockLeftRight | 表示左右摆动 | M   |
| 1   | RockUpDown    | 表示上下摆动 | M   |
| 2   | RockRound     | 指示环绕摆动 | M   |

##### 6.5.4.5.2 WindBitmap 类型

WindBitmap 类型包含以下数据字段，详见下表270。

表270 WindBitmap 类型表

| Bit | 名称          | 概述    | 一致性 |
|-----|-------------|-------|-----|
| 0   | SleepWind   | 表示睡眠风 | M   |
| 1   | NaturalWind | 表示自然风 | M   |

其中，

- SleepWind字段：风扇的速度基于当前设置，应逐渐减慢至最低速度。对于此过程，序列、速度和持续时间是MS。
- NaturalWind字段：风扇的速度应变化以模拟自然风。对于此设置，序列、速度和持续时间是MS。

##### 6.5.4.5.3 StepDirectionEnum 类型

StepDirectionEnum 类型包含以下数据字段，详见下表271。

表271 StepDirectionEnum 类型表

| Bit | 名称       | 概述         | 一致性 |
|-----|----------|------------|-----|
| 0   | Increase | 步进以增加的方向移动 | M   |
| 1   | Decrease | 步进以减少的方向移动 | M   |

#### 6.5.4.5.4 AirflowDirectionEnum 类型

AirflowDirectionEnum 类型包含以下数据字段，详见下表272。

表272 AirflowDirectionEnum 类型表

| Bit | 名称      | 概述   | 一致性 |
|-----|---------|------|-----|
| 0   | Forward | 气流向前 | M   |
| 1   | Reverse | 气流反向 | M   |

#### 6.5.4.5.5 FanModeEnum 类型

AirflowDirectionEnum 类型包含以下数据字段，详见下表273。

表273 AirflowDirectionEnum 类型表

| Bit | 名称     | 概述       | 一致性  |
|-----|--------|----------|------|
| 0   | Off    | 风扇关闭     | M    |
| 1   | Low    | 风扇使用低速   | desc |
| 2   | Medium | 风扇使用中速   | desc |
| 3   | High   | 风扇使用高速   | M    |
| 4   | On     | —        | D    |
| 5   | Auto   | 风扇使用自动模式 | AUT  |
| 6   | Smart  | 风扇使用智能模式 | D    |

——Low 字段

如果风扇支持2种或更多速度，则应支持Low字段。当且仅当FanModeSequence属性值小于4时，Low字段应得到支持。

——Medium 字段

如果风扇支持3种或更多速度，则应支持Medium字段。当且仅当FanModeSequence属性值为0或2时，Medium字段应得到支持。

#### 6.5.4.5.6 FanModeSequenceEnum 类型

FanMode SequenceEnum 类型包含以下数据字段，详见下表274。

表274 FanMode SequenceEnum 类型表

| 值 | 名称                | 概述                   | 一致性      |
|---|-------------------|----------------------|----------|
| 0 | OffLowMedHigh     | 风扇具有关闭、低速、中速和高速模式    | [!AUT].a |
| 1 | OffLowHigh        | 风扇具有关闭、低速和高速模式       | [!AUT].a |
| 2 | OffLowMedHighAuto | 风扇具有关闭、低速、中速、高速和自动模式 | [!AUT].a |
| 3 | OffLowHighAuto    | 风扇具有关闭、低速、高速和自动模式    | [!AUT].a |
| 4 | OffHighAuto       | 风扇具有关闭、高速和自动模式       | [!AUT].a |
| 5 | OffHigh           | 风扇具有关闭和高速模式          | [!AUT].a |

#### 6.5.4.6 属性

##### 6.5.4.6.1 属性列表

风扇控制功能集属性见下表275。

表275 风扇控制功能集属性表

| 标识符    | 名称               | 类型                   | 约束条件        | 质量等级 | 默认   | 通道     | 一致性    |
|--------|------------------|----------------------|-------------|------|------|--------|--------|
| 0x0000 | FanMode          | FanModeEnum          | all         | N    | 0    | RWVO   | M      |
| 0x0001 | FanModeSequence  | FanModeSequenceEnum  | all         | N    | MS   | R[W]VO | Zigbee |
| 0x0001 | FanModeSequence  | FanModeSequenceEnum  | all         | F    | MS   | RV     | M      |
| 0x0002 | PercentSetting   | percent              | max100      | X    | 0    | RWVO   | M      |
| 0x0003 | PercentCurrent   | percent              | max100      | —    | desc | RV     | M      |
| 0x0004 | SpeedMax         | uint8                | 1to100      | F    | MS   | RV     | SPD    |
| 0x0005 | SpeedSetting     | uint8                | maxSpeedMax | X    | 0    | RWVO   | SPD    |
| 0x0006 | SpeedCurrent     | uint8                | maxSpeedMax | P    | desc | RV     | SPD    |
| 0x0007 | RockSupport      | RockBitmap           | desc        | F    | 0    | RV     | RCK    |
| 0x0008 | RockSetting      | RockBitmap           | desc        | P    | 0    | RWVO   | RCK    |
| 0x0009 | WindSupport      | WindBitmap           | desc        | F    | 0    | RV     | WND    |
| 0x000A | WindSetting      | WindBitmap           | desc        | P    | 0    | RWVO   | WND    |
| 0x000B | AirflowDirection | AirflowDirectionEnum | desc        | P    | 0    | RWVO   | DIR    |

#### 6.5.4.6.2 FanMode 属性

##### 6.5.4.6.2.1 概述

此属性应指示风扇的当前速度模式。客户端可以写入此属性以请求不同的风扇模式。服务端应返回 INVALID\_IN\_STATE，以指示风扇当前处于无法更改FanMode的状态。服务端可能有某些FanMode值，这些值永远无法设置。例如，当此功能集与具有系统依赖性的其他功能集（例如Thermostat功能集）出现在同一端点或另一个端点上时，尝试将此功能集的FanMode属性设置为Off可能不被系统允许。

此属性应设置为FanModeEnum中的一个值。

当成功写入FanMode属性时，PercentSetting和SpeedSetting（如果存在）属性应设置为适当值，具体定义见第6.6.4.7.4节和第6.6.4.7.7节，除非另有说明。

当FanMode属性设置为任何给定模式时，PercentCurrent和SpeedCurrent（如果存在）应当指示实际当前运行的风扇速度，除非下文另有规定。

##### 6.5.4.6.2.2 Off 值

将属性值设置为Off应当将这些属性的值设置为0（零）：

- PercentSetting
- PercentCurrent
- SpeedSetting（如果存在）
- SpeedCurrent（如果存在）

##### 6.5.4.6.2.3 Auto 值

将属性值设置为Auto应当将这些属性的值设置为null：

- PercentSetting
- SpeedSetting（如果存在）

当此属性值为Auto时，这些属性应当继续指示风扇的当前状态：

- PercentCurrent
- SpeedCurrent（如果存在）

##### 6.5.4.6.2.4 On 值

如果客户端尝试写入On的值，属性应当设置为High。

##### 6.5.4.6.2.5 Smart 值

如果客户端尝试写入Smart的值并且支持AUT功能，属性应当设置为Auto，否则属性应当设置为High。

#### 6.5.4.6.3 FanModeSequence 属性

此属性指示应受支持的风扇速度范围。

#### 6.5.4.6.4 PercentSetting 属性

此属性应指示风扇的速度设置。此属性可由客户端写入以指示新的风扇速度。如果客户端对此属性写入null,属性值不应改变。服务端应返回INVALID\_IN\_STATE以指示风扇未处于可将PercentSetting更改为请求值的状态。

如果成功写入0,服务端应将FanMode属性值设置为Off。

——Percent 规则

由服务端实现来映射PercentSetting属性和FanMode属性枚举值之间的范围。百分比值被划分为范围,每个范围对应一个受支持的FanMode属性值。百分比范围不应重叠。如果支持,高速范围内的所有百分比值都应高于中速和低速范围内的所有百分比值。中速范围内的所有百分比值都应高于低速范围内的所有百分比值。如果客户端将FanMode属性设置为低、中或高,服务端应如果客户端设置了PercentSetting属性,服务端应根据百分比值在对应范围内将FanMode属性设置为低、中或高。

如果支持MultiSpeed特性,从百分比值变化计算SpeedSetting或SpeedCurrent(速度)对于PercentSetting或PercentCurrent(百分比)应保持真实,如公式(12)所示:

$$\text{速度} = \text{向上取整}(\text{SpeedMax} * (\text{百分比} * 0.01)) \dots\dots\dots (12)$$

示例:如果 SpeedMax 属性为 42(42 速风扇),并且 PercentSetting 更改为 25,则 SpeedSetting 和 SpeedCurrent 变为 11(10.5 向上取整)。

#### 6.5.4.6.5 PercentCurrent 属性

此属性应指示实际当前运行的风扇速度,或者为零以指示风扇已关闭。由于其他系统要求不允许风扇按照请求的设置运行,此属性的值和PercentSetting属性的值之间可能存在暂时的不匹配。

#### 6.5.4.6.6 SpeedMax 属性

此属性应指示风扇有一个速度(值为1)或最大速度,如果风扇能够以多个速度运行。

#### 6.5.4.6.7 SpeedSetting 属性

此属性应指示风扇的速度设置。此属性可由客户端写入以指示新的风扇速度。如果客户端对此属性写入null,属性值不应改变。服务端应返回INVALID\_IN\_STATE以指示风扇未处于可以将SpeedSetting更改为请求值的状态。

如果成功写入0,服务端应将FanMode属性值设置为Off。

——Speed 规则

将SpeedSetting属性和FanMode属性枚举值之间的映射关系留给服务端实现来决定。速度值被划分为多个范围,每个范围对应一个FanMode属性值。速度范围不得重叠。如果支持,高速范围内的所有速度值应高于中速和低速范围内的所有速度值。中速范围内的所有速度值应高于低速范围内的所有速度值。如果客户端将FanMode属性设置为低、中或高,服务端应将SpeedSetting属性设置为对应范围内的值。如果客户端设置SpeedSetting属性,服务端应根据速度值在对应范围内将FanMode属性设置为低、中或高。

从速度值变化计算PercentSetting或PercentCurrent(百分比)的值对于SpeedSetting或SpeedCurrent(速度)的计算公式必须成立,详见公式(13)。

$$\text{PercentCurrent} = \text{floor} \dots\dots\dots (13)$$

示例:如果 SpeedMax 属性为 42(42 速风扇),并且 SpeedSetting 属性更改为 11,则 PercentSetting 和 PercentCurrent 变为 26。

#### 6.5.4.6.8 SpeedCurrent 属性

此属性应指示实际当前运行的风扇速度,或零以指示风扇关闭。由于其他系统要求不允许风扇在请求的设置下运行,此属性的值和SpeedSetting属性的值之间可能存在暂时的不匹配。

#### 6.5.4.6.9 RockSupport 属性

此属性是一个位图,指示服务端支持的摇摆动作。

#### 6.5.4.6.10 RockSetting 属性

此属性是一个位图，指示当前活动的风扇摇摆动作设置。仅当RockSupport属性中相应的位设置为1时，每个位才应设置为1，否则应返回状态代码CONSTRAINT\_ERROR。

如果客户端设置的受支持位的组合不被服务端支持，则组合中受支持的最低单个位应被设置并处于活动状态，而所有其他位应表示零。

示例：如果同时设置了 RockUpDown 和 RockRound，但这种组合是不可能的，那么只有 RockUpDown 变为活动状态。

#### 6.5.4.6.11 WindSupport 属性

此属性是一个位图，指示服务端支持的风力模式。至少一个风模式位应当被设置。

#### 6.5.4.6.12 WindSetting 属性

此属性是一个位图，指示当前活动的风扇风力特性设置。如果WindSupport属性中对应的位设置为1，则每个位仅应设置为1，否则应返回状态码CONSTRAINT\_ERROR。

如果客户端设置的受支持位的组合不被服务端支持，则组合中受支持的最低单个位应被设置并处于活动状态，而所有其他位应表示零。

示例：如果设置了睡眠风和自然风，但这种组合不可能实现，那么只有睡眠风会变得活跃。

#### 6.5.4.6.13 AirflowDirection 属性

此属性应指示风扇的当前空气流向。此属性可由客户端写入，以指示风扇的新空气流向。此属性应设置为AirflowDirectionEnum表中的一个值。

#### 6.5.4.7 命令

风扇控制功能集命令见下表276。

表276 风扇控制功能集命令表

| 标识符  | 名称   | 方向            | 返回 | 通道 | 一致性  |
|------|------|---------------|----|----|------|
| 0x00 | Step | client→server | Y  | 0  | STEP |

Step命令可以加快或减慢风扇的速度，以步进方式，而无需客户端了解风扇速度。此命令支持，例如，用户操作的墙壁开关，其中用户提供反馈或控制，以在达到适当速度时停止发送此命令。步进速度值是实现特定的。实施了多少步进速度也是实现特定的。

此命令支持字段信息详见下表277。

表277 Step 命令数据字段表

| 标识符 | 名称        | 类型                | 约束条件 | 质量等级 | 默认       | 一致性 |
|-----|-----------|-------------------|------|------|----------|-----|
| 0   | Direction | StepDirectionEnum | —    | —    | Increase | M   |
| 1   | Wrap      | bool              | —    | —    | false    | 0   |
| 2   | LowestOff | bool              | —    | —    | true     | 0   |

其中，

- Direction字段应指示风扇速度是增加还是减少到下一个步进值。
- Wrap字段应指示风扇速度是否在最高和最低步进值之间循环。
- LowestOff字段应指示风扇关闭（速度值为0）是否被包含为步进值。
- 生成时间：客户端发送此命令以逐步加快或减慢风扇速度。

即使风扇速度当前不在实现的步进值上，也应执行此命令。

如果Direction字段为增加，

- 如果风扇速度低于最高步进值，风扇速度应更改为比当前风扇速度高的最低步进值。
- 否则，如果Wrap为TRUE，风扇速度应更改为最低步进值。
- 否则，风扇速度应更改为（或保持在）最高步进值。

如果Direction字段为减少，

- 如果风扇速度高于最低步进值，风扇速度应变为低于当前风扇速度的最高步进值。
- 否则，如果Wrap为TRUE，风扇速度应变为最高步进值。
- 否则，风扇速度应变为（或保持在）最低步进值。

尽管Step命令的效果是实现特定的，但收到Step命令后对受影响属性的效果应遵循一致性。

6.5.5 恒温用户界面配置功能集

6.5.5.1 概述

本功能集提供了一个接口，允许配置支持键盘和LCD屏幕的恒温器或恒温器控制器设备的用户界面。

6.5.5.2 分类

恒温用户界面配置功能集分类详见下表278。

表278 恒温用户界面配置功能集分类

| 等级   | 角色          | 范围       | PICS 编码 |
|------|-------------|----------|---------|
| Base | Application | Endpoint | TSUIC   |

6.5.5.3 Cluster ID

恒温用户界面配置功能集ID详见下表279。

表279 恒温用户界面配置功能集 ID

| 标识符    | 名称        |
|--------|-----------|
| 0x0204 | 恒温器用户界面配置 |

6.5.5.4 温度值的显示转换

参见数据模型中的温度转换部分，了解华氏度和摄氏度之间的单位转换。

6.5.5.5 数据类型

6.5.5.5.1 TemperatureDisplayModeEnum 类型

此数据类型源自enum8，详见下表280。

表280 ITemperatureDisplayModeEnum 类型

| Bit | 名称  | 概述       | 一致性 |
|-----|-----|----------|-----|
| 0   | 摄氏度 | 以℃显示的温度  | M   |
| 1   | 华氏度 | 以°F显示的温度 | M   |

6.5.5.5.2 KeypadLockoutEnum 类型

此数据类型源自enum8，详见下表281。

表281 KeypadLockoutEnum 类型

| Bit | 名称  | 概述         | 一致性 |
|-----|-----|------------|-----|
| 0   | 无锁定 | 所有可供用户使用功能 | M   |
| 1   | 锁定1 | 第1级减少功能    | M   |
| 2   | 锁定2 | 第2级减少功能    | M   |
| 3   | 锁定3 | 第3级减少功能    | M   |
| 4   | 锁定4 | 第4级减少功能    | M   |
| 5   | 锁定5 | 对用户提供的最少功能 | M   |

各种级别的解释取决于设备。

6.5.5.5.3 ScheduleProgrammingVisibilityEnum 类型

此数据类型源自enum8，详见下表282。

表282 ScheduleProgramingVisibilityEnum 类型

| Bit | 名称                           | 概述              | 一致性 |
|-----|------------------------------|-----------------|-----|
| 0   | ScheduleProgrammingPermitted | 恒温器上启用了本地计划编程功能 | M   |
| 1   | ScheduleProgrammingDenied    | 恒温器上禁用了本地计划编程功能 | M   |

6.5.5.6 属性

恒温用户界面配置功能集属性见下表283。

表283 恒温用户界面配置功能集属性表

| 标识符    | 名称                            | 类型                                | 约束条件 | 质量等级 | 默认                           | 通道   | 一致性 |
|--------|-------------------------------|-----------------------------------|------|------|------------------------------|------|-----|
| 0x0000 | TemperatureDisplayMode        | TemperatureDisplayModeEnum        | all  | –    | Celsius                      | RWVO | M   |
| 0x0001 | KeypadLockout                 | KeypadLockoutEnum                 | all  | –    | NoLockout                    | RWVM | M   |
| 0x0002 | ScheduleProgrammingVisibility | ScheduleProgrammingVisibilityEnum | all  | –    | ScheduleProgrammingPermitted | RWVM | O   |

其中，

- TemperatureDisplayMode属性应指示恒温器屏幕上显示的温度单位。
- KeypadLockout属性应指示通过键盘对用户可用的功能级别。
- ScheduleProgrammingVisibility属性用于向用户隐藏恒温器上的每周计划编程功能或菜单，以防止用户在恒温器上进行本地编程设置每周计划。通过远程界面仍可以执行调度编程，恒温器可以在调度编程模式下运行。此属性旨在防止用户或服务提供商通过更强大的远程界面设置的计划被本地篡改或禁用。即使用户在恒温器上看不到，程序计划也将继续运行。

7 照明相关设备类型库

7.1 基本设备类型

7.1.1 概述

如下定义了所有设备类型的通用一致性，具体取决于但不限于：

- 认证程序（例如Zigbee、Matter等）
- 底层协议栈（例如802.15.4、Wi-Fi、Thread、ZigbeePRO、IPv6、TCP/IP）
- 地区法规
- 界面（UI、云服务等）
- 范围（例如住宅与商业）
- 其他常见限制或功能（例如电池供电或休眠节点）
- 其他等等

7.1.2 条件说明

7.1.2.1 概述

以下每个部分都是条件的一个类别，每个类别定义了一系列符合条件的名称和唯一标签。分类的划分仅供阅读用途。

7.1.2.2 认证程序情况

在本章节首次发布时，许多认证计划已经终止，或仅允许重新认证，例如ZHA标准。详见下表284。

表284 认证程序情况汇总表

| 认证程序                 | 标签     | 描述                 |
|----------------------|--------|--------------------|
| ZigbeeHomeAutomation | ZHA    | 家庭自动化标准            |
| ZigbeeSmartEnergy    | ZSE    | zigbee智能能源标准       |
| GreenPower           | GP     | ZigbeeGreenPower标准 |
| Zigbee               | Zigbee | Zigbee标准           |
| SuZi                 | SuZi   | ZigbeePROSub-GHz标准 |
| Matter               | Matter | Matter标准           |

7.1.2.3 协议情况

基本设备类型的协议情况汇总，详见下表 285。

表285 协议情况汇总表

| 协议标签     |
|----------|
| Ethernet |
| Wi-Fi    |
| Thread   |
| TCP      |
| UDP      |
| IP       |
| IPv4     |
| IPv6     |

7.1.2.4 人机交互情况

基本设备类型的人机交互情况汇总，详见下表 286。

表286 人机交互情况汇总表

| 交互标签           | 描述                 |
|----------------|--------------------|
| LanguageLocale | 该节点支持本地化的向用户传递文本   |
| TimeLocale     | 该节点支持本地化的向用户传递时间   |
| UnitLocale     | 该节点支持本地化的向用户传递度量单位 |

注：上表中的“支持本地化”指的是通过功能集交互支持本地化更新。

7.1.3 通用能力情况

基本设备类型适用于节点的常见限制或功能，详见下表287。

表287 通用能力情况

| 能力标签   | 描述                 |
|--------|--------------------|
| SIT    | 节点是一个短暂空闲间歇性可连接的设备 |
| LIT    | 节点是一个长期空闲间歇性可连接的设备 |
| Active | 节点一直处于可通信状态        |

7.1.4 设备类型类别情况

此类别用于设备类型的分类。其中一些分类取决于其他条件。详见下表288。

表288 设备类型类别情况

| 类别标签     | 概述                                      |
|----------|-----------------------------------------|
| Node     | 设备类型被归类为Node设备类型（参见“5.6 数据模型规范”）        |
| App      | 设备类型被归类为Application设备类型（参见“5.6 数据模型规范”） |
| Simple   | 设备类型被归类为Simple设备类型（参见“5.6 数据模型规范”）      |
| Dynamic  | 设备类型被归类为Dynamic设备类型（参见“5.6 数据模型规范”）     |
| Composed | 设备类型由2种或更多的设备类型组成（参见“5.6 数据模型规范”）       |

7.1.5 端点型号类型情况

这个类别用于端点的分类。其中一些分类依赖于其他条件。详见下表289。

表289 端点型号类型情况

| 类别标签                   | 概述                        |
|------------------------|---------------------------|
| Client                 | 端点上的一个客户端功能集              |
| Server                 | 端点上的一个服务器功能集              |
| Duplicate              | 详见重复条件                    |
| BridgedPowerSourceInfo | 该端点代表一个桥接设备，其电源状态信息可供网桥使用 |

——重复条件

该端点及其至少一个兄弟端点在应用设备类型上有重叠，如系统模型规范中的“Disambiguation”部分所定义。这种情况触发了提供有关端点的附加信息的要求，以便在端点之间消除歧义。

7.1.6 功能集需求

每个Matter设备类型的实现都应该至少包含这些功能集，且基于下面定义的一致性。这个一致性表应假设Matter一致性条件是TRUE（在一致性列中）。详见下表290。

表290 功能集需求表

| 标识符    | 功能集名称      | 客户端/服务端 | 质量等级 | 一致性           |
|--------|------------|---------|------|---------------|
| 0x001D | Descriptor | Server  | –    | M             |
| 0x001E | Binding    | Server  | –    | Simple&Client |
| 0x0040 | FixedLabel | Server  | –    | 0             |
| 0x0041 | UserLabel  | Server  | –    | 0             |

7.1.7 元素需求

下表291列出了覆盖功能集规范要求的品质和一致性。空白项意味着没有变化。

表291 元素需求

| 标识符    | 功能集名称      | 元素      | 名称      | 约束条件 | 通道 | 一致性       |
|--------|------------|---------|---------|------|----|-----------|
| 0x001D | Descriptor | Feature | TagList |      | –  | Duplicate |

7.2 功用类设备类型

7.2.1 功用类设备类型汇总表

本章介绍功用设备的类型，功用类设备类型汇总如下表292所示。

表292 功用设备类型汇总表

| 设备标识符  | 设备名称              |
|--------|-------------------|
| 0x0016 | RootNode（根节点）     |
| 0x0011 | PowerSource（供电来源） |
| 0x0012 | OTA请求者            |
| 0x0014 | OTA提供者            |
| 0x0013 | BridgedNode（桥接设备） |
| 0x0510 | 电传感器              |
| 0x050D | 设备电源管理            |
| 0x0019 | 辅助网络接口            |
| 0x0130 | 联合Fabric管理员       |

7.2.2 根节点

7.2.2.1 概述

这定义了根节点端点的一致性（见“5.8 系统模型规范”）。这个端点类似于一个“readmefirst”的端点，它描述了自身以及构成节点的其他端点。

- 具有端点范围的设备类型不应与该设备类型支持在同一个端点上。
- 具有应用角色的功能集不应与该设备类型支持在同一个端点上。
- 其他具有节点范围的设备类型可以与该设备类型支持在同一个端上。

7.2.2.2 类别

功用类设备类型根节点类别如下表 293。

表293 类别

| 标识符    | 设备名称     | 上层类别 | 分类   | 范围   |
|--------|----------|------|------|------|
| 0x0016 | RootNode | -    | Node | Node |

7.2.2.3 条件

功用类设备类型根节点条件要求如下表 294。

表294 条件

| 条件                  | 描述                                                                                           |
|---------------------|----------------------------------------------------------------------------------------------|
| CustomNetworkConfig | 该节点仅支持带外配置网络（例如，具有丰富的用户界面方式、制造商指定的方式、自定义配网流程，或者采用还没有直接由NetworkCommissioning功能集支持的将来IP兼容网络技术） |
| ManagedAclAllowed   | 节点至少有一个端点上的设备类型具有将此条件设置为true的设备类型库元素要求表条目                                                    |

请参阅“7.1 基本设备类型”定义以获得额外的conformancetags。

7.2.2.4 设备类型要求

该表列出了根据一致性要与该设备类型一起实现的其他设备类型。详见下表295。

表295 设备类型要求

| 标识符    | 名称          | 约束条件 | 一致性 |
|--------|-------------|------|-----|
| 0x0011 | PowerSource | -    | 0   |

7.2.2.5 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表296。

表296 根节点功能集要求

| 标识符    | 功能集                        | 客户端/服务端u | 质量等级 | 一致性                  |
|--------|----------------------------|----------|------|----------------------|
| 0x0028 | BasicInformation           | Server   | I    | M                    |
| 0x001F | AccessControl              | Server   | I    | M                    |
| 0x002E | PowerSourceConfiguration   | Server   | I    | O, D                 |
| 0x0038 | TimeSynchronization        | Server   | I    | O                    |
| 0x003F | GroupKeyManagement         | Server   | I    | M                    |
| 0x0030 | GeneralCommissioning       | Server   | I    | M                    |
| 0x0031 | NetworkCommissioning       | Server   | —    | !CustomNetworkConfig |
| 0x003C | AdministratorCommissioning | Server   | I    | M                    |
| 0x003E | NodeOperationalCredentials | Server   | I    | M                    |
| 0x002B | LocalizationConfiguration  | Server   | I    | LanguageLocale       |
| 0x002C | TimeFormatLocalization     | Server   | I    | TimeLocale           |
| 0x002D | UnitLocalization           | Server   | I    | UnitLocale           |
| 0x0033 | GeneralDiagnostics         | Server   | I    | M                    |
| 0x0032 | DiagnosticLogs             | Server   | I    | O                    |
| 0x0034 | SoftwareDiagnostics        | Server   | I    | O                    |
| 0x0037 | EthernetNetworkDiagnostics | Server   | —    | [Ethernet]           |
| 0x0036 | Wi-FiNetworkDiagnostics    | Server   | —    | [Wi-Fi]              |
| 0x0035 | ThreadNetworkDiagnostics   | Server   | —    | [Thread]             |
| 0x0046 | ICDManagement              | Server   | I    | SIT LIT              |

注：根节点上的NetworkDiagnosticscluster应服务于NetworkCommissioningcluster（如果存在）中指定的主网络接口或带外配置的网络接口。

### 7.2.2.6 元素要求

下表列出了覆盖功能集规范要求的Quality和Conformance。空白的表格单元格意味着该项目和来自功能集规范里的值没有发生变化。详见下表297。

表297 功能集规范元素要求

| 标识符    | 功能集名称          | 元素      | 名称                  | 约束条件 | 通道 | 一致性                 |
|--------|----------------|---------|---------------------|------|----|---------------------|
| 0x001F | AccessControl  | Feature | MNGD                | desc | —  | [ManagedAclAllowed] |
| 0x0046 | ICD Management | Feature | LongIdleTimeSupport | —    | —  | LIT                 |

——AccessControlMNGD 符合性

设备根节点端点的访问控制集群（AccessControlCluster）的 MNGD（管理设备，ManagedDevice）功能仅限于包含明确允许使用此功能的应用端点类型的设备，例如网络基础设施管理器设备类型（设备类型 ID 0x0090）。

### 7.2.2.7 端点组合

根节点端点的Descriptor功能集的PartsList属性应包含节点上所有其他端点的列表，例如，“5.8 系统模型规范”中定义的所有模型。

### 7.2.3 供电来源

#### 7.2.3.1 类别

供电来源类别信息详见下表298。

表298 类别

| 标识符    | 设备名称 | 上层类别 | 分类      | 范围   |
|--------|------|------|---------|------|
| 0x0011 | 供电来源 | —    | Utility | Node |

#### 7.2.3.2 功能集要求

此设备应支持下表中列出的clusters。详见下表399。

表299 供电来源功能集要求

| 标识符    | 功能集         | 客户端/服务端 | 质量等级 | 一致性 |
|--------|-------------|---------|------|-----|
| 0x002F | PowerSource | Server  | -    | M   |

#### 7.2.4 OTA 请求者

##### 7.2.4.1 概述

OTA请求者是一个能够接收OTA软件更新的设备。

##### 7.2.4.2 类别

这是OTA请求者的类别表，详见表300。

表300 OTA 请求者类别表

| 标识符    | 设备名称   | 上层类别 | 分类      | 范围   |
|--------|--------|------|---------|------|
| 0x0012 | OTA请求者 | -    | Utility | Node |

##### 7.2.4.3 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表301。

表301 OTA 请求者功能集要求

| 标识符    | 功能集                        | 客户端/服务端 | 质量等级 | 一致性 |
|--------|----------------------------|---------|------|-----|
| 0x002A | OTASoftwareUpdateRequestor | Server  | -    | M   |
| 0x0029 | OTASoftwareUpdateProvider  | Client  | -    | M   |

#### 7.2.5 OTA 提供者

##### 7.2.5.1 概述

OTA提供者是一个能够向同一fabric上的其他节点提供OTA软件更新的节点。

##### 7.2.5.2 类别

这是OTA提供者的类别信息，详见表302。

表302 OTA 提供者类别表

| 标识符    | 设备名称   | 上层类别 | 分类      | 范围   |
|--------|--------|------|---------|------|
| 0x0014 | OTA提供者 | -    | Utility | Node |

##### 7.2.5.3 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。一个节点最多只能拥有一个OTA提供者所需clusters的实例。详见下表303。

表303 OTA 提供者功能集要求

| 标识符    | 功能集                        | 客户端/服务端 | 质量等级 | 一致性 |
|--------|----------------------------|---------|------|-----|
| 0x002A | OTASoftwareUpdateRequestor | Client  | -    | O   |
| 0x0029 | OTASoftwareUpdateProvider  | Server  | -    | M   |

#### 7.2.6 桥接设备

##### 7.2.6.1 概述

该章节定义了桥接节点根端点的一致性。这个端点类似于“readmefirst”端点，它描述自己和组成桥接设备的任何其他端点。桥接节点的根端点表示外部网络上的设备，但不是桥接节点本身的根端点。

### 7.2.6.2 类别

这是桥接设备类别的类别信息，详见表304。

表304 桥接设备类别表

| 标识符    | 设备名称        | 上层类别 | 分类      | 范围       |
|--------|-------------|------|---------|----------|
| 0x0013 | BridgedNode | —    | Utility | Endpoint |

### 7.2.6.3 条件

此设备类型可能支持以下定义的符合性条件。详见下表305。

表305 桥接设备条件要求表

| 条件                     | 描述                                 |
|------------------------|------------------------------------|
| FabricSynchronizedNode | 参见以下“FabricSynchronizedNode条件”的描述。 |

请参阅“7.1 基本设备类型”定义的其他符合性标签。

——FabricSynchronizedNode 条件

当以下所有条件都为真时，FabricSynchronizedNode条件适用于桥接节点端点：

- 以该端点为子端点的Aggregator有一个CommissionerControlCluster。
- 该 CommissionerControlCluster 有一个 SupportedDeviceCategories 属性 设置 了 FabricSynchronization位。
- 该桥接节点是一个Matter节点。

### 7.2.6.4 设备类型要求

此设备类型应仅在具有Aggregator设备类型的另一个端点的DescriptorclusterPartsList中列出的端点上进行标识。

该表306列出了根据符合性要求与该设备类型一起实现的其他设备类型。

表306 桥接设备设备类型要求

| 标识符    | 名称          | 约束条件 | 一致性 |
|--------|-------------|------|-----|
| 0x0011 | PowerSource | —    | 0   |

### 7.2.6.5 功能集要求

基于以下一致性的定义，支持该桥接设备类型的每个端点应包括这些clusters。详见下表307。

表307 桥接设备功能集要求

| 标识符    | 功能集                           | 客户端/服务端 | 质量等级 | 一致性                       |
|--------|-------------------------------|---------|------|---------------------------|
| 0x0039 | BridgedDeviceBasicInformation | Server  | —    | M                         |
| 0x002E | PowerSourceConfiguration      | Server  | —    | BridgedPowerSourceInfo, D |
| 0x002F | PowerSource                   | Server  | —    | BridgedPowerSourceInfo    |
| 0x0750 | EcosystemInformation          | Server  | —    | FabricSynchronizedNode, 0 |
| 0x003C | AdministratorCommissioning    | Server  | —    | FabricSynchronizedNode    |

### 7.2.6.6 端点组合

桥接设备端点应支持以下组合模式之一：

- 分散的端点：所有应用设备类型都应在单独的端点上受支持，而不是在桥接设备端点上。桥接节点端点的Descriptor功能集的PartsList属性应指示表示桥接设备功能的所有端点列表，包括支持应用设备类型的端点，即“5.8 系统模型规范”中定义的所有模式。
- 单个端点：桥接节点和一个或多个应用设备类型应在同一端点上受支持（遵循应用设备类型规则）。端点组合应符合应用设备类型（一个或多个）的定义。

### 7.2.7 电气传感器

7.2.7.1 概述

电气传感器设备用于测量输入输出的电功率和电量。

7.2.7.2 类别

这是电气传感器设备的类别信息，详见表308。

表308 电气传感器设备类别

| 标识符    | 设备名称 | 上层类别 | 分类      | 范围       |
|--------|------|------|---------|----------|
| 0x0510 | 电传感器 | -    | Utility | Endpoint |

7.2.7.3 设备类型要求

由ElectricalPowerMeasurement功能集、ElectricalEnergyMeasurement功能集或两者进行的电气测量应适用于PowerTopology功能集所指示的端点。

7.2.7.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表309。

表309 电气传感器设备功能集要求

| 标识符    | 名称                          | 客户端/服务端 | 质量等级 | 一致性  |
|--------|-----------------------------|---------|------|------|
| 0x009C | PowerTopology               | Server  | -    | M    |
| 0x0090 | ElectricalPowerMeasurement  | Server  | -    | 0.a+ |
| 0x0091 | ElectricalEnergyMeasurement | Server  | -    | 0.a+ |

7.2.8 设备电源管理

7.2.8.1 概述

设备能源管理装置提供报告，并可选地调整计划由该设备消耗或产生的电功耗。

7.2.8.2 类别

这是设备电源管理的类别信息，详见表310。

表310 设备电源管理类别

| 标识符    | 设备名称   | 上层类别 | 分类      | 范围       |
|--------|--------|------|---------|----------|
| 0x050D | 设备电源管理 | -    | Utility | Endpoint |

7.2.8.3 条件

请参阅“7.1 基本设备类型”定义的符合性标签。详见下表311。

表311 条件

| 条件              | 描述                          |
|-----------------|-----------------------------|
| ControllableESA | 该端点上的DEMcluster接受命令来调整其能源运行 |

其中，ControllableESA设备允许客户端请求更改功率（PowerAdjustment特性），更改开始时间（StartTimeAdjustment特性），暂停和恢复（可暂停特性），或调整其功率或状态预测（ForecastAdjustment或ConstraintBasedAdjustment特性）。

将预测作为单一功能进行简单报告（支持PowerForecastReporting或stateforecastreporting特性）不要求ControllableESA条件。

7.2.8.4 cluster 要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表312。

表312 设备能源管理 clusters 要求

| 标识符    | 名称                         | 客户端/服务端 | 质量等级 | 一致性                |
|--------|----------------------------|---------|------|--------------------|
| 0x0098 | DeviceEnergyManagement     | Server  | —    | M                  |
| 0x009F | DeviceEnergyManagementMode | Server  | —    | ControllableESA, 0 |

### 7.2.8.5 元素要求

下面列出了覆盖cluster规范要求的特征和符合性。空白的表单元格意味着没有对该项进行更改，并且采用cluster规范中的值。详见下表313。

表313 设备能源管理元素要求

| 标识符    | 功能集                    | 元素      | 名称                        | 约束条件 | 通道 | 一致性                    |
|--------|------------------------|---------|---------------------------|------|----|------------------------|
| 0x0098 | DeviceEnergyManagement | Feature | PowerAdjustment           | —    | —  | ControllableESA.<br>a+ |
| 0x0098 | DeviceEnergyManagement | Feature | StartTimeAdjustment       | —    | —  | ControllableESA.<br>a+ |
| 0x0098 | DeviceEnergyManagement | Feature | Pausable                  | —    | —  | ControllableESA.<br>a+ |
| 0x0098 | DeviceEnergyManagement | Feature | ForecastAdjustment        | —    | —  | ControllableESA.<br>a+ |
| 0x0098 | DeviceEnergyManagement | Feature | ConstraintBasedAdjustment | —    | —  | ControllableESA.<br>a+ |

### 7.2.9 辅助网络接口

#### 7.2.9.1 概述

辅助网络接口设备提供节点支持的附加网络接口，以补充由根节点端点托管的主接口。支持多个网络接口的节点应包括在根节点端点上的主接口，以及其他端点上的辅助接口。这些网络接口的优先级由其端点的顺序决定，端点号越小的接口优先级越高。

#### 7.2.9.2 类别

以下是辅助网络接口设备的类别信息，详见表314。

表314 辅助网络接口设备类别

| 标识符    | 设备名称                               | 上层类别 | 分类           | 范围            |
|--------|------------------------------------|------|--------------|---------------|
| 0x0019 | 辅助网络接口 (SecondaryNetworkInterface) | —    | 功用 (Utility) | 端点 (Endpoint) |

#### 7.2.9.3 cluster 要求

支持该设备类型的每个端点应根据以下定义的符合性包含这些cluster。详见下表315。

表315 辅助网络接口设备 cluster 要求

| 标识符    | 名称                         | 客户端/服务端 | 质量等级 | 一致性        |
|--------|----------------------------|---------|------|------------|
| 0x0031 | NetworkCommissioning       | Server  | —    | M          |
| 0x0037 | EthernetNetworkDiagnostics | Server  | —    | [Ethernet] |
| 0x0036 | Wi-FiNetworkDiagnostics    | Server  | —    | [Wi-Fi]    |
| 0x0035 | ThreadNetworkDiagnostics   | Server  | —    | [Thread]   |

注：该设备类型中的NetworkDiagnosticscluster应服务于NetworkCommissioningcluster中指定的辅助网络接口。

### 7.3 应用类设备类型

以下章节列出了在本文件的“Matter设备库”中定义了应用类设备类型。它们按每一章的功能区域分组并且概述了相关的设备类型如下表316所示。

表316 应用设备类型汇总表

| 标识符              | 设备名称             |
|------------------|------------------|
| 照明类设备类型          |                  |
| 0x010D           | 可调颜色和色温的增强型灯     |
| 智能插头/插座等执行器类设备类型 |                  |
| 0x010A           | 智能插座             |
| 0x010B           | 可调节供电等级的插座设备     |
| 0x010F           | 固定式开/关控制         |
| 0x0110           | 固定式可调光负载控制       |
| 0x0303           | 泵                |
| 0x0042           | 水阀               |
| 开关控制类设备类型        |                  |
| 0x0103           | 支持开/关灯的开关        |
| 0x0104           | 支持调光的开关          |
| 0x0105           | 支持调光和切换颜色的开关     |
| 0x0840           | 支持开关调光、颜色以及情景的枢纽 |
| 0x0304           | 泵控制器             |
| 0x000F           | 通用开关             |
| 传感器类设备类型         |                  |
| 0x0015           | 门磁/接触传感器         |
| 0x0106           | 光照度测量传感器         |
| 0x0107           | 占位传感器            |
| 0x0302           | 温度传感器            |
| 0x0305           | 压力传感器            |
| 0x0307           | 湿度传感器            |
| 0x0850           | 开关传感器            |
| 0x0076           | 烟雾警报器            |
| 0x002C           | 空气质量传感器          |
| 0x0041           | 水冰点检测            |
| 0x0043           | 水泄漏检测            |
| 0x0044           | 雨水传感器            |
| 暖通空调类设备类型        |                  |
| 0x0301           | 恒温器              |
| 0x002B           | 风扇               |
| 0x002D           | 空气净化器            |

后期spec会增加新的设备ID以及大类，后续文档会继续更新。

如果申请新增设备类型，通常需要提交MRD申请，可参考该文档<https://groups.csa-iot.org/wg/members/document/21624>。

## 7.4 照明类设备类型

### 7.4.1 开关灯

#### 7.4.1.1 概述

开关灯（On/OffLight）是一种能够通过绑定的控制器设备（如开/关灯开关（On/OffLightSwitch）或调光开关（DimmerSwitch））打开或关闭的照明设备。此外，开关灯也能够通过绑定占用传感器进行切换。

#### 7.4.1.2 类别

以下是开关灯设备的类别信息，详见表317。

表317 开关灯设备类别

| 标识符    | 设备名称        | 上层类别 | 类别     | 范围       |
|--------|-------------|------|--------|----------|
| 0x0100 | On/OffLight | —    | Simple | Endpoint |

#### 7.4.1.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

#### 7.4.1.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集，详见下表318。

表318 开关灯功能集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | —    | M    |
| 0x0004 | Groups           | Server  | —    | M    |
| 0x0062 | ScenesManagement | Server  | —    | P, M |
| 0x0006 | On/Off           | Server  | —    | M    |
| 0x0008 | LevelControl     | Server  | —    | O    |
| 0x0406 | OccupancySensing | Client  | —    | O    |

建议在此设备上包含级别控制功能集（LevelControlcluster），以便在设备与其他可调光灯分组并使用“带开关”命令时提供一致的用户体验。对于此设备，由于其唯一的状态是开或关，因此如果实现了级别控制功能集，它不应影响实际的光亮级别，除非是那些导致开/关状态变化的命令，即“带开关”命令。此外，如果实现了级别控制功能集，设备应接受并处理级别控制功能集命令，相应地调整CurrentLevel属性的值，并在必要时调整开/关功能集的On/Off属性。

#### 7.4.1.5 元素要求

下表319列出了覆盖功能集规范要求的特性和一致性。空白的单元格表示对该项没有更改而是使用功能集规范中的值。

表319 功能集元素要求

| 标识符    | 功能集              | 元素        | 名称            | 约束条件     | 通道 | 一致性  |
|--------|------------------|-----------|---------------|----------|----|------|
| 0x0003 | Identify         | Command   | TriggerEffect | —        | —  | M    |
| 0x0062 | ScenesManagement | Command   | CopyScene     | —        | —  | P, M |
| 0x0006 | On/Off           | Feature   | Lighting      | —        | —  | M    |
| 0x0008 | LevelControl     | Feature   | OnOff         | —        | —  | M    |
| 0x0008 | LevelControl     | Feature   | Lighting      | —        | —  | M    |
| 0x0008 | LevelControl     | Attribute | CurrentLevel  | 1 to 254 | —  | —    |
| 0x0008 | LevelControl     | Attribute | MinLevel      | 1        | —  | —    |
| 0x0008 | LevelControl     | Attribute | MaxLevel      | 254      | —  | —    |

由于Identify功能集的TriggerEffect命令和On/Off功能集的OffWithEffect命令指定的灯光效果需要调光输出，而此类设备无法实现调光功能，因此指定的灯光效果可以用纯开关灯光效果替代。

### 7.4.2 可调光灯

#### 7.4.2.1 概述

可调光灯（DimmableLight）是一种能够通过绑定的控制器设备（如调光开关（DimmerSwitch）或调色开关（ColorDimmerSwitch））打开或关闭、调整光强度的照明设备。此外，可调光灯也能够通过绑定占用传感器或其他设备进行切换。

#### 7.4.2.2 类别

以下是可调光灯设备的类别信息，详见表320。

表320 可调光类别

| 标识符    | 设备名称          | 上层类别        | 分类     | 范围       |
|--------|---------------|-------------|--------|----------|
| 0x0101 | DimmableLight | On/OffLight | Simple | Endpoint |

#### 7.4.2.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

#### 7.4.2.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集，详见下表321。

表321 可调光灯功能集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | —    | M    |
| 0x0004 | Groups           | Server  | —    | M    |
| 0x0062 | ScenesManagement | Server  | —    | P, M |
| 0x0006 | On/Off           | Server  | —    | M    |
| 0x0008 | LevelControl     | Server  | —    | M    |
| 0x0406 | OccupancySensing | Client  | —    | 0    |

#### 7.4.2.5 元素要求

下面列出了覆盖功能集规范要求的特性和一致性。空白的单元格表示对该项没有更改而是使用功能集规范中的值。详见下表322。

表322 功能集元素要求

| 标识符    | 功能集              | 元素        | 名称            | 约束条件   | 通道 | 一致性  |
|--------|------------------|-----------|---------------|--------|----|------|
| 0x0003 | Identify         | Command   | TriggerEffect | —      | —  | M    |
| 0x0062 | ScenesManagement | Command   | CopyScene     | —      | —  | P, M |
| 0x0006 | On/Off           | Feature   | Lighting      | —      | —  | M    |
| 0x0008 | LevelControl     | Feature   | OnOff         | —      | —  | M    |
| 0x0008 | LevelControl     | Feature   | Lighting      | —      | —  | M    |
| 0x0008 | LevelControl     | Attribute | CurrentLevel  | 1to254 | —  | —    |
| 0x0008 | LevelControl     | Attribute | MinLevel      | 1      | —  | —    |
| 0x0008 | LevelControl     | Attribute | MaxLevel      | 254    | —  | —    |

#### 7.4.3 色温灯

##### 7.4.3.1 概述

色温灯（ColorTemperatureLight）是一种能够通过绑定的控制器设备（如调色开关（ColorDimmerSwitch））打开或关闭、调整光强度和调整色温的照明设备。

##### 7.4.3.2 类别

以下是色温灯的类别信息，详见表323。

表323 色温灯类别

| 标识符    | 设备名称                  | 上层类别          | 分类     | 范围       |
|--------|-----------------------|---------------|--------|----------|
| 0x010C | ColorTemperatureLight | DimmableLight | Simple | Endpoint |

##### 7.4.3.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

##### 7.4.3.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集。详见下表324。

表324 色温灯功能集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | —    | M    |
| 0x0004 | Groups           | Server  | —    | M    |
| 0x0062 | ScenesManagement | Server  | —    | P, M |
| 0x0006 | On/Off           | Server  | —    | M    |
| 0x0008 | LevelControl     | Server  | —    | M    |
| 0x0300 | ColorControl     | Server  | —    | M    |
| 0x0406 | OccupancySensing | Client  | —    | O    |

#### 7.4.3.5 元素要求

下面列出了覆盖功能集规范要求的特性和一致性。空白的单元格表示对该项没有更改而是使用功能集规范中的值。详见下表325。

表325 元素要求

| 标识符    | 功能集              | 元素        | 名称               | 约束条件   | 通道 | 一致性  |
|--------|------------------|-----------|------------------|--------|----|------|
| 0x0003 | Identify         | Command   | TriggerEffect    | —      | —  | M    |
| 0x0062 | ScenesManagement | Command   | CopyScene        | —      | —  | P, M |
| 0x0006 | On/Off           | Feature   | Lighting         | —      | —  | M    |
| 0x0008 | LevelControl     | Feature   | OnOff            | —      | —  | M    |
| 0x0008 | LevelControl     | Feature   | Lighting         | —      | —  | M    |
| 0x0008 | LevelControl     | Attribute | CurrentLevel     | 1to254 | —  | —    |
| 0x0008 | LevelControl     | Attribute | MinLevel         | 1      | —  | —    |
| 0x0008 | LevelControl     | Attribute | MaxLevel         | 254    | —  | —    |
| 0x0300 | ColorControl     | Feature   | ColorTemperature | —      | —  | M    |
| 0x0300 | ColorControl     | Attribute | RemainingTime    | —      | —  | M    |

### 7.4.4 增强型颜色灯

#### 7.4.4.1 概述

增强型颜色灯（ExtendedColorLight）是一种能够通过绑定的控制器设备（如调色开关（ColorDimmerSwitch）或控制台（ControlBridge））打开或关闭、调整光强度和调整颜色的照明设备。设备支持通过色相/饱和度、增强色相、颜色循环、XY坐标和色温的方式调整颜色。此外，增强型颜色灯也能够通过绑定占用传感器进行切换。

#### 7.4.4.2 类别

以下是增强型颜色灯的类别信息，详见表326。

表326 增强型颜色灯类别

| 标识符    | 设备名称               | 上层类别                  | 分类     | 范围       |
|--------|--------------------|-----------------------|--------|----------|
| 0x010D | ExtendedColorLight | ColorTemperatureLight | Simple | Endpoint |

#### 7.4.4.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

#### 7.4.4.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集。详见下表327。

表327 增强型颜色灯功能集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | —    | M    |
| 0x0004 | Groups           | Server  | —    | M    |
| 0x0062 | ScenesManagement | Server  | —    | P, M |
| 0x0006 | On/Off           | Server  | —    | M    |
| 0x0008 | LevelControl     | Server  | —    | M    |
| 0x0300 | ColorControl     | Server  | —    | M    |
| 0x0406 | OccupancySensing | Client  | —    | O    |

#### 7.4.4.5 元素要求

下面列出了覆盖功能集规范要求的特性和一致性。空白的单元格表示对该项没有更改而是使用功能集规范中的值。详见下表328。

表328 元素要求

| 标识符    | 功能集              | 元素        | 名称               | 约束条件   | 通道 | 一致性  |
|--------|------------------|-----------|------------------|--------|----|------|
| 0x0003 | Identify         | Command   | TriggerEffect    | —      | —  | M    |
| 0x0062 | ScenesManagement | Command   | CopyScene        | —      | —  | P, M |
| 0x0006 | On/Off           | Feature   | Lighting         | —      | —  | M    |
| 0x0008 | LevelControl     | Feature   | OnOff            | —      | —  | M    |
| 0x0008 | LevelControl     | Feature   | Lighting         | —      | —  | M    |
| 0x0008 | LevelControl     | Attribute | CurrentLevel     | 1to254 | —  | —    |
| 0x0008 | LevelControl     | Attribute | MinLevel         | 1      | —  | —    |
| 0x0008 | LevelControl     | Attribute | MaxLevel         | 254    | —  | —    |
| 0x0300 | ColorControl     | Feature   | HueSaturation    | —      | —  | O    |
| 0x0300 | ColorControl     | Feature   | EnhancedHue      | —      | —  | O    |
| 0x0300 | ColorControl     | Feature   | ColorLoop        | —      | —  | O    |
| 0x0300 | ColorControl     | Feature   | xy               | —      | —  | M    |
| 0x0300 | ColorControl     | Feature   | ColorTemperature | —      | —  | M    |
| 0x0300 | ColorControl     | Attribute | RemainingTime    | —      | —  | M    |

### 7.5 智能插头/插座等执行器类设备类型

#### 7.5.1 智能插座

##### 7.5.1.1 概述

开/关插座设备是一种能够通过绑定的控制器设备（如开/关灯遥控开关或调光遥控开关）来打开或关闭的设备，开/关插座设备通常用于控制传统的非通信灯，也可以通过这种方式进行控制其他电器，例如非智能的烧水壶、饮水机等。

##### 7.5.1.2 类别

以下是智能插座的类别信息，详见表329。

表329 智能插座类别

| 标识符    | 设备名称 | 上层类别 | 分类     | 范围       |
|--------|------|------|--------|----------|
| 0x010A | 智能插座 | —    | Simple | Endpoint |

##### 7.5.1.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

7.5.1.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集。详见下表330。

表330 智能插头/插座等执行器类设备类型的功能集需求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | –    | M    |
| 0x0004 | Groups           | Server  | –    | M    |
| 0x0062 | ScenesManagement | Server  | –    | P, M |
| 0x0006 | On/Off           | Server  | –    | M    |
| 0x0008 | LevelControl     | Server  | –    | O    |
| 0x0406 | OccupancySensing | Client  | –    | O    |

建议在此设备上包含级别/等级控制cluster，提升用户一致的体验，当设备与其他的可调光灯分组时，并接收到了“withon/off”命令，对于这个设备，由于它只有开或关的状态，如果该设备实现了级别/等级控制功能集的规范，除了那些导致开/关状态变化的命令（即“withon/off”命令外），它不会对实际输出的电平产生任何影响（因为只有开关状态）。另外，如果实施了LevelControl功能集，设备需要接受并处理LevelControl功能集的相关命令，需根据响应的值来调整CurrentLevel属性值，条件满足时也调整On/Off功能集的OnOff属性。

7.5.1.5 元素要求

下面列出了覆盖功能集规范要求的特性和一致性。空白的单元格表示对该项和来自功能集规范中的值没有更改。详见下表331。

表331 元素要求

| 标识符    | 功能集              | 元素        | 名称            | 约束条件   | 通道 | 一致性  |
|--------|------------------|-----------|---------------|--------|----|------|
| 0x0003 | Identify         | Command   | TriggerEffect | –      | –  | M    |
| 0x0062 | ScenesManagement | Command   | CopyScene     | –      | –  | P, M |
| 0x0006 | On/Off           | Feature   | Lighting      | –      | –  | M    |
| 0x0008 | LevelControl     | Feature   | OnOff         | –      | –  | M    |
| 0x0008 | LevelControl     | Feature   | Lighting      | –      | –  | M    |
| 0x0008 | LevelControl     | Attribute | CurrentLevel  | 1to254 | –  | –    |
| 0x0008 | LevelControl     | Attribute | MinLevel      | 1      | –  | –    |
| 0x0008 | LevelControl     | Attribute | MaxLevel      | 254    | –  | –    |

由于Identify功能集的TriggerEffect命令和On/Off功能集的offwiththeeffect命令指定了需要调暗光输出的光效，而该功能这在此设备类型上是不可能的，因此指定的光效可能被纯开/关光效所取代。

7.5.2 可调节供电等级的插座

7.5.2.1 概述

可调节供电等级的插座除了有智能插座的功能外，可以支持供电等级调整的功能，如通过被绑定到调光开关，调色开关来被控制该设备，该设备控制的原理为通过切减交流电的相位来调整输出驱动等级。

7.5.2.2 类别

以下是可调节供电等级的插座的类别信息，详见表332。

表332 可调节供电等级的插座类别

| 标识符    | 设备名称       | 上层类别 | 分类     | 范围       |
|--------|------------|------|--------|----------|
| 0x010B | 可调节供电等级的插座 | –    | Simple | Endpoint |

7.5.2.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

#### 7.5.2.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集。详见下表333。

表333 智能插座功能集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | —    | M    |
| 0x0004 | Groups           | Server  | —    | M    |
| 0x0062 | ScenesManagement | Server  | —    | P, M |
| 0x0006 | On/Off           | Server  | —    | M    |
| 0x0008 | LevelControl     | Server  | —    | M    |
| 0x0406 | OccupancySensing | Client  | —    | 0    |

#### 7.5.2.5 元素要求

下面列出了覆盖cluster规范要求的特性和一致性。空白的单元格表示对该项没有更改，并且应用cluster规范中的值。详见下表334。

表334 元素要求

| 标识符    | 功能集              | 元素        | 名称            | 约束条件   | 通道 | 一致性  |
|--------|------------------|-----------|---------------|--------|----|------|
| 0x0003 | Identify         | Command   | TriggerEffect | —      | —  | M    |
| 0x0062 | ScenesManagement | Command   | CopyScene     | —      | —  | P, M |
| 0x0006 | On/Off           | Feature   | Lighting      | —      | —  | M    |
| 0x0008 | LevelControl     | Feature   | OnOff         | —      | —  | M    |
| 0x0008 | LevelControl     | Feature   | Lighting      | —      | —  | M    |
| 0x0008 | LevelControl     | Attribute | CurrentLevel  | 1to254 | —  | —    |
| 0x0008 | LevelControl     | Attribute | MinLevel      | 1      | —  | —    |
| 0x0008 | LevelControl     | Attribute | MaxLevel      | 254    | —  | —    |

### 7.5.3 固定式开/关控制

#### 7.5.3.1 概述

固定式开/关控制是一种固定安装的装置，可为另一个插入其中的设备提供电源，并能够切换该供电的开或关。

#### 7.5.3.2 分类

以下是固定式开/关控制的类别信息，详见表335。

表335 固定式开/关控制类别

| 标识符    | 设备名称                 | 上层类别 | 分类     | 范围       |
|--------|----------------------|------|--------|----------|
| 0x010F | MountedOn/OffControl | —    | Simple | Endpoint |

#### 7.5.3.3 条件

请参阅“7.1 基本设备类型”定义以了解符合性标签。

#### 7.5.3.4 功能集要求

每个端点在支持此设备类型时，应根据以下定义的符合性来包含这些功能集。详见下表336。

表336 固定式开/关控制集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | –    | M    |
| 0x0004 | Groups           | Server  | –    | M    |
| 0x0062 | ScenesManagement | Server  | –    | P, M |
| 0x0006 | On/Off           | Server  | –    | M    |
| 0x0008 | LevelControl     | Server  | –    | O    |
| 0x0406 | OccupancySensing | Client  | –    | O    |

建议在此设备上包含LevelControlcluster，以在与其他可调光灯分组并使用“withon/off”命令时提供一致的用户体验。对于此设备，由于它仅有开或关两种状态，如果实现了LevelControlcluster，则除会导致开/关状态变化的命令（即“withon/off”命令）外，不得对实际亮度产生影响。此外，若实现了LevelControlcluster，该设备必须接收并处理该cluster的命令，适当调整CurrentLevel属性，并在必要时调整On/Offcluster的OnOff属性。

### 7.5.3.5 元素要求

以下列出了覆盖Cluster规范要求的质量和一致性。空白的表格单元格表示该项目没有任何变化，并采用Cluster规范中的默认值。详见下表337。

表337 元素要求

| 标识符    | 功能集              | 元素        | 名称            | 约束条件   | 通道 | 一致性  |
|--------|------------------|-----------|---------------|--------|----|------|
| 0x0003 | Identify         | Command   | TriggerEffect | –      | –  | M    |
| 0x0062 | ScenesManagement | Command   | CopyScene     | –      | –  | P, M |
| 0x0006 | On/Off           | Feature   | Lighting      | –      | –  | M    |
| 0x0008 | LevelControl     | Feature   | OnOff         | –      | –  | M    |
| 0x0008 | LevelControl     | Feature   | Lighting      | –      | –  | M    |
| 0x0008 | LevelControl     | Attribute | CurrentLevel  | 1to254 | –  | –    |
| 0x0008 | LevelControl     | Attribute | MinLevel      | 1      | –  | –    |
| 0x0008 | LevelControl     | Attribute | MaxLevel      | 254    | –  | –    |

由于Identify功能集的TriggerEffect命令和On/Off功能集的OffWithEffect命令指定了需要调暗灯光输出的灯光效果，而此设备类型无法实现该功能，所以这些指定的灯光效果可使用纯开/关灯光效果替代。

## 7.5.4 固定式可调光负载控制

### 7.5.4.1 概述

固定式可调光负载控制是一种固定安装的装置，可为插入其中的另一个设备提供电源，并能够开启或关闭该电源并调整其输出等级。可调光负载控制通常通过电源的相位切割方式来控制传统的非通信照明设备。

### 7.5.4.2 分类

以下是固定式可调光负载控制的类别信息，详见表338。

表338 固定式可调光负载控制类别

| 标识符    | 设备名称                       | 上层类别 | 分类     | 范围       |
|--------|----------------------------|------|--------|----------|
| 0x0110 | MountedDimmableLoadControl | –    | Simple | Endpoint |

### 7.5.4.3 条件

请参阅“7.1 基本设备类型”定义以了解一致性标签。

### 7.5.4.4 功能集要求

每个支持此设备类型的端点应根据下面定义的符合性包含这些功能集。详见下表339。

表339 固定式可调光负载控制功能集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | —    | M    |
| 0x0004 | Groups           | Server  | —    | M    |
| 0x0062 | ScenesManagement | Server  | —    | P, M |
| 0x0006 | On/Off           | Server  | —    | M    |
| 0x0008 | LevelControl     | Server  | —    | M    |
| 0x0406 | OccupancySensing | Client  | —    | O    |

#### 7.5.4.5 元素要求

下面列出了覆盖功能集规范要求的特性和一致性。空白的单元格表示对该项没有更改，并且应用功能集规范中的值。详见下表340。

表340 元素要求

| 标识符    | 功能集              | 元素        | 名称            | 约束条件   | 通道 | 一致性  |
|--------|------------------|-----------|---------------|--------|----|------|
| 0x0003 | Identify         | Command   | TriggerEffect | —      | —  | M    |
| 0x0062 | ScenesManagement | Command   | CopyScene     | —      | —  | P, M |
| 0x0006 | On/Off           | Feature   | Lighting      | —      | —  | M    |
| 0x0008 | LevelControl     | Feature   | OnOff         | —      | —  | M    |
| 0x0008 | LevelControl     | Feature   | Lighting      | —      | —  | M    |
| 0x0008 | LevelControl     | Attribute | CurrentLevel  | 1to254 | —  | —    |
| 0x0008 | LevelControl     | Attribute | MinLevel      | 1      | —  | —    |
| 0x0008 | LevelControl     | Attribute | MaxLevel      | 254    | —  | —    |

### 7.6 开关控制类设备类型

#### 7.6.1 概述

本章指定了许多“控制器”设备类型，如On/Off Light开关和调光开关。实现这些设备类型的一些产品旨在取代传统开关或者直接控制负载功率的调光器。对于这类产品，鼓励制造商实施额外的端点在同一产品上持有“致动器”设备类型，如开/关灯（或开/关插座）或可调光灯（或可调光插座），与它可以提供给负载的控制类型一致。在产品可以单独控制多路负载的情况下，多个这样的端点为每个负载持有一个设备类型。

此外，拥有中央控制功能可以实现更丰富的自动化触发的通过按下开关按键。在这种情况下，开关的工作原理更像是传感器，为此，定义了通用开关设备类型。查看通用开关，鼓励制造商在通常被称为开关的产品中实现通用开关设备类型，查看开关设备类型了解这些设备类型如何组合的示例。

#### 7.6.2 无线开关

##### 7.6.2.1 概述

无线开关是一种控制器设备，当绑定到照明设备（如可被开/关的灯）时，能用于控制打开或关闭设备。

##### 7.6.2.2 类别

以下是无线开关的类别信息，详见表341。

表341 无线开关类别

| 标识符    | 设备名称              | 上层类别 | 分类     | 范围       |
|--------|-------------------|------|--------|----------|
| 0x0103 | On/OffLightSwitch | —    | Simple | Endpoint |

##### 7.6.2.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

### 7.6.2.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集。详见下表342。

表342 无线开关功能集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | —    | M    |
| 0x0003 | Identify         | Client  | —    | M    |
| 0x0004 | Groups           | Client  | —    | O    |
| 0x0006 | On/Off           | Client  | —    | M    |
| 0x0062 | ScenesManagement | Client  | —    | P, O |

### 7.6.3 无线调光灯控制开关

#### 7.6.3.1 概述

调光开关是一个控制器设备，当绑定到一个照明装置时可以用来无线控制该设备，如控制增强型的彩色灯，能够被用来调整灯的亮度。

#### 7.6.3.2 类别

以下是无线调光控制开关的类别信息，详见表343。

表343 无线调光控制开关类别

| 标识符    | 设备名称         | 上层类别              | 分类     | 范围       |
|--------|--------------|-------------------|--------|----------|
| 0x0104 | DimmerSwitch | On/OffLightSwitch | Simple | Endpoint |

#### 7.6.3.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

#### 7.6.3.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集。详见下表344。

表344 无线调光控制开关功能集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | —    | M    |
| 0x0003 | Identify         | Client  | —    | M    |
| 0x0004 | Groups           | Client  | —    | O    |
| 0x0062 | ScenesManagement | Client  | —    | P, O |
| 0x0006 | On/Off           | Client  | —    | M    |
| 0x0008 | LevelControl     | Client  | —    | M    |

### 7.6.4 无线彩灯控制开关

#### 7.6.4.1 概述

五线彩灯控制开关是一个控制器设备，当绑定到一个照明装置时可以用来无线控制该设备，如控制增强型的彩色灯，能够被用来调整灯的颜色与亮度。

#### 7.6.4.2 类别

以下是五线彩灯控制开关的类别信息，详见表345。

表345 五线彩灯控制开关类别

| 标识符    | 设备名称              | 上层类别         | 分类     | 范围       |
|--------|-------------------|--------------|--------|----------|
| 0x0105 | ColorDimmerSwitch | DimmerSwitch | Simple | Endpoint |

#### 7.6.4.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

#### 7.6.4.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集。详见下表346。

表346 五线彩灯控制开关功能集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | –    | M    |
| 0x0003 | Identify         | Client  | –    | M    |
| 0x0004 | Groups           | Client  | –    | O    |
| 0x0062 | ScenesManagement | Client  | –    | P, O |
| 0x0006 | On/Off           | Client  | –    | M    |
| 0x0008 | LevelControl     | Client  | –    | M    |
| 0x0300 | ColorControl     | Client  | –    | M    |

#### 7.6.5 支持开关调光、颜色以及情景的枢纽

##### 7.6.5.1 概述

控制桥是一种控制器设备，当它与照明设备（例如ExtendedColorLight）绑定时，可以用来开关设备、调整发出光的强度以及调整发出光的颜色。此外，控制桥设备还可以用来设置场景。

##### 7.6.5.2 类别

以下是控制桥的类别信息，详见表347。

表347 控制桥类别

| 标识符    | 设备名称          | 上层类别 | 分类     | 范围       |
|--------|---------------|------|--------|----------|
| 0x0840 | ControlBridge |      | Simple | Endpoint |

##### 7.6.5.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

##### 7.6.5.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集。详见下表348。

表348 控制桥功能集要求

| 标识符    | 功能集                    | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------------|---------|------|------|
| 0x0003 | Identify               | Server  | –    | M    |
| 0x0003 | Identify               | Client  | –    | M    |
| 0x0004 | Groups                 | Client  | –    | M    |
| 0x0062 | ScenesManagement       | Client  | –    | P, M |
| 0x0006 | On/Off                 | Client  | –    | M    |
| 0x0008 | LevelControl           | Client  | –    | M    |
| 0x0300 | ColorControl           | Client  | –    | M    |
| 0x0400 | IlluminanceMeasurement | Client  | –    | O    |
| 0x0406 | OccupancySensing       | Client  | –    | O    |

#### 7.6.6 通用开关

##### 7.6.6.1 概述

该章节定义了通用开关设备类型的一致性。

##### 7.6.6.2 类别

以下是通用开关的类别信息，详见表349。

表349 通用开关类别

| 标识符    | 设备名称          | 上层类别 | 分类     | 范围       |
|--------|---------------|------|--------|----------|
| 0x000F | GenericSwitch | —    | Simple | Endpoint |

### 7.6.6.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

### 7.6.6.4 功能集要求

#### 7.6.6.4.1 功能集要求概述

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集。详见下表350。

表350 通用开关功能集要求

| 标识符    | 功能集      | 客户端/服务端 | 质量等级 | 一致性 |
|--------|----------|---------|------|-----|
| 0x0003 | Identify | Server  | —    | M   |
| 0x003B | Switch   | Server  | —    | M   |

#### 7.6.6.4.2 即时上报

订阅和事件的通用机制可能不能确保检测到的开关交互将被“即时”发送到感兴趣的一方的switch客户端cluster（它们可能只在一段时间后发送，例如由于批事件行为和订阅的“MinInterval”行为）。

为获得良好的用户体验，该设备类型需及时向订阅方发送Switchcluster中定义的属性和事件的更新。

#### 7.6.6.4.3 对多个开关的设备标示

在Matter协议中，包含多个开关的节点需要暴露多个端点，每个端点都承载这种设备类型和关联的Switch功能集的一个实例。这意味着Matter基础设备要求中的Duplicate条件适用，因此每个此类端点的Descriptor功能集中都应包含一个TagList。这个TagList中的标签用于表示与用户相关的方向（例如，两个按钮的左右）或标记（例如，印在按钮上的“调亮”和“调暗”图标）。客户端应使用这些标签向用户传达此类信息（例如，在用户界面中显示），以帮助用户识别哪个端点对应于某个方向或标记。

对于服务器指示来自CommonNumberNamespace的标签的情况，以及客户端呈现与端点相关的实体（例如，各种开关的图标），它应按标签指示的数值顺序呈现它们，如CommonNumberNamespace所示。

对于只有一个端点承载这种设备类型和关联的Switch功能集的节点，可以使用TagList。在开关具有用户可识别的标记的情况下，这可能会有益。

TagList可以包含来自MatterSemanticTagNamespaces中定义的命名空间的组合，包括开关的命名空间以及来自制造商特定命名空间的标签。

如果按钮有预期的功能（例如，雕刻的图标），则应使用来自如果没有相应的标签，应使用带有字符串标签的制造商特定标签（见下文示例2）。

为了识别设备上按钮的位置（例如，两个按钮设备的顶部按钮），应使用来自对于这些不适用或不足的设备（例如，一行四个按钮的开关设备），应使用来自从左到右，从上到下的顺序，从第一个按钮开始编号。

在适用的情况下，为了使设备能够控制闭合类设备（Closure），应使用通用闭合命名空间中的语义标签。示例1：一个设备上有两个并列的摇杆开关，它有2个端点（11，12）用于与开关相关的功能。

- 端点11的设备类型为通用开关和包含：
  - Switch功能集（功能标志：LS），显示左键的状态和事件
  - Descriptor功能集的TagList包含两个标签：Position.Left和Number.One
- 端点12的设备类型为通用开关和包含：
  - Switch功能集（功能标志：LS），显示右键的状态和事件
  - Descriptor功能集的TagList包含两个标签：Position.Right和Number.Two

如果这个设备的按钮上有“向上”和“向下”的图标标记，那么TagList将包含第三个标签（来自Switches命名空间），分别为Switches.Up和Switches.Down。

示例2：一个设备有四个按钮（安装在一个正方形中），每个按钮都有一个特定场景设置的图标，它有四个端点（21，22，23，24），用于开关相关的功能。

- 端点21的设备类型为通用开关和包含：
  - Switch功能集（特性标志：MS）显示左上角按钮的事件
  - Descriptor功能集的TagList包含四个标签：Position.Top, Position.Left, Number.One以及（标签=Switches.Custom, 标签=“watchtv”）
    - ◆ 这个最后的标签是一个带有标签（其他三个标签不需要Label字段）的Switches.Custom标签
- 端点22的设备类型为通用开关和包含：
  - Switch功能集（特性标志：MS）显示右上角按钮的事件
  - Descriptor功能集的TagList包含四个标签：Position.Top, Position.Right, Number.Two以及（标签=Switches.Custom, 标签=“dinner”）
- 端点23的设备类型为通用开关和包含：
  - Switch功能集（特性标志：MS）显示左下角按钮的事件
  - Descriptor功能集的TagList包含四个标签：Position.Bottom, Position.Left, Number.Three以及（标签=Switches.Custom, 标签=“reading”）
- 端点24的设备类型为通用开关和包含：
  - Switch功能集（特性标志：MS）显示右下角按钮的事件
  - Descriptor功能集的TagList包含四个标签：Position.Bottom, Position.Right, Number.Four以及（标签=Switches.Custom, 标签=“nightlight”）

#### 7.6.6.5 与其他开关设备类型的关系（参考）

通用开关设备类型和开/关灯无线开关设备类型都传递了与一个开关交互信息到另一个设备。

- 开/关灯开关将从其开/关（客户端）cluster发送开/关/切换命令到实现开/关（服务端）cluster的设备，以控制设备的开/关功能，开/关灯开关设备还可以实现组和场景cluster，从而发送组和场景命令，基本上，它是直接向其他目标设备发送控制命令，其中由绑定表告诉设备需要将命令发送到哪里。
- 通用开关设备类型将向实现开关客户端cluster的订阅者发送属性（仅用于LatchingSwitch）和事件的更新，作为与开关交互的指示——将该解释（例如，对应的设备应该被执行由于通讯交互）留给订阅方，所以它可以被比作传感器式的设备，这允许一个更全面的控制器将来自开关的信息与其他输入或者信息源（例如，一天的时间，用户存在于该区域）结合起来，以确定哪些控制命令（例如，开/关，场景执行，属性改变）发送到网络中的其他设备。

一个设备制造商可以在同一个开关设备上实现两种类型的设备，以允许它被用于两种类型的控制，就像这个例子中的摇杆开关实现：

- 端点31的设备类型为开关和包含：
  - （客户端）clusterOn/Off输出On/Off/Toggle命令
- 端点32的设备类型为通用开关和包含：
  - （服务端）cluster开关（特性标志：LS），提示当前开关的状态和事件，便于其他客户端获取。

当该设备在特定设置中使用，可以使用绑定表和订阅来确定它使用的是哪种方式：

- 用作开/关灯开关（32号终端不需要订阅）
- 用作通用开关（端点31不需要绑定）
- 同时用作两者。

在这种情况下，与开关的交互将导致一个On/Off/Toggle命令被发送到端点31的绑定表中存在的设备，以及属性更新和事件被发送到端点32的订阅设备。

## 7.7 传感器类设备类型

### 7.7.1 门磁传感器设备类型

#### 7.7.1.1 概述

该章节定义了设备类型为门磁的一致性规范。

#### 7.7.1.2 类别

以下是门磁传感器设备的类别信息，详见表351。

表351 门磁传感器设备类别

| 标识符    | 设备名称          | 上层类别 | 分类     | 范围       |
|--------|---------------|------|--------|----------|
| 0x0015 | ContactSensor | —    | Simple | Endpoint |

#### 7.7.1.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

#### 7.7.1.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表352。

表352 门磁传感器设备功能集要求

| 标识符    | 名称                        | 客户端/服务端 | 质量等级 | 一致性 |
|--------|---------------------------|---------|------|-----|
| 0x0003 | Identify                  | Server  | —    | M   |
| 0x0045 | BooleanState              | Server  | —    | M   |
| 0x0080 | BooleanStateConfiguration | Server  | —    | 0   |

——定义 Cluster

这个被用于定义端点。

——布尔状态 Cluster

这用来指示传感器/探测器的状态。

布尔状态Cluster的状态应反映使用此方案进行传感器检测。详见下表353。

表353 门磁传感器设备功能集布尔状态

| 值     | 状态              |
|-------|-----------------|
| True  | Closedorcontact |
| False | Openornocontact |

——布尔状态配置 Cluster

这用于配置传感器/探测器，对于这种设备类型来说，它与布尔状态Cluster的配置相关联。

#### 7.7.1.5 元素限制

下面列出了覆盖cluster规范要求的特性和一致性。空白的单元格表示对该项没有更改，并且应用cluster规范中的值。详见下表354。

表354 元素限制

| 标识符 | 功能集      | 元素      | 名称    | 约束条件 | 通道 | 一致性     |
|-----|----------|---------|-------|------|----|---------|
| 0   | Identify | Feature | Query | —    | —  | !Matter |

## 7.7.2 光照度传感器

### 7.7.2.1 概述

光传感装置是一种测量和传感设备，它能够测量和报告光的强度（照度）到需要接受的一方。

### 7.7.2.2 类别

以下是光传感装置的类别信息，详见表355。

表355 光传感装置类别

| 标识符    | 设备名称        | 上层类别 | 分类     | 范围       |
|--------|-------------|------|--------|----------|
| 0x0106 | LightSensor | -    | Simple | Endpoint |

### 7.7.2.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

### 7.7.2.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表356。

表356 光传感装置功能集要求

| 标识符    | 功能集                    | 客户端/服务端 | 质量等级 | 一致性 |
|--------|------------------------|---------|------|-----|
| 0x0003 | Identify               | Server  | -    | M   |
| 0x0004 | Groups                 | Client  | -    | O   |
| 0x0400 | IlluminanceMeasurement | Server  | -    | M   |

### 7.7.2.5 元素限制

下面列出了覆盖cluster规范要求的特性和一致性。空白的单元格表示对该项没有更改，并且应用cluster规范中的值。详见下表357。

表357 元素限制

| 标识符 | 功能集      | 元素      | 名称    | 约束条件 | 通道 | 一致性     |
|-----|----------|---------|-------|------|----|---------|
| 0   | Identify | Feature | Query | -    | -  | !Matter |

### 7.7.3 占位传感器设备类型

#### 7.7.3.1 概述

占位传感器是一种测量和传感设备，它能够测量和报告指定区域的居住状态。

#### 7.7.3.2 类别

以下是占位传感器的类别信息，详见表358。

表358 占位传感器类别

| 标识符    | 设备名称            | 上层类别 | 分类     | 范围       |
|--------|-----------------|------|--------|----------|
| 0x0107 | OccupancySensor | -    | Simple | Endpoint |

#### 7.7.3.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

#### 7.7.3.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表359。

表359 占位传感器功能集要求

| 标识符    | 名称                        | 客户端/服务端 | 质量等级 | 一致性 |
|--------|---------------------------|---------|------|-----|
| 0x0003 | Identify                  | Server  | -    | M   |
| 0x0080 | BooleanStateConfiguration | Server  | -    | O   |
| 0x0406 | OccupancySensing          | Server  | -    | M   |

——定义 Cluster

这是被用于定义端点。

——布尔状态配置 Cluster

这用于配置传感器/探测器，并且与占用感应功能集的配置相关联。

——占位传感器 Cluster

这用于指示占用情况以及用于检测和配置有关占用和未占用转换的占用传感器的类型。

——多模态传感器

占用感应功能集定义了多种可用于检测占用的模态。

支持多种模态（在特性标志中体现）的设备可以通过以下两种方式实现：

- 单个端点实现一个占用感应功能集，并设置两个或更多特性位为1。
  - 需要将组合后的感应结果作为一个比特位上报到Occupancy属性（以及在支持的情况下，通过OccupancyChanged事件），并使用同一组定时参数。
  - 灵敏度设置（通过位于同一端点的布尔状态配置功能集实现）会以制造商特定的映射方式同时应用于所有感应模态。
- 多个端点，每个端点都带有一个占用感应功能集（每个端点只设置一个特性位）：
  - 每种模态的感应结果会分别上报到各自端点的Occupancy属性（以及在支持的情况下，通过OccupancyChanged事件），由该端点中功能集提供的一组定时参数进行管理。
  - 这意味着某些端点中的这些属性可能与其他端点的对应属性值不同，如果客户端想要获得单一结果，则可能需要自行合并这些值。
  - 可以通过一个或多个端点上所提供的布尔状态配置功能集，为每种模态提供独立的灵敏度设置。

7.7.4 温度传感器

7.7.4.1 概述

温度传感器是用于测量当前温度的设备，并且将发送到接收方。

7.7.4.2 类别

以下是温度传感器的类别信息，详见表360。

表360 温度传感器类别

| 标识符    | 设备名称              | 上层类别 | 分类     | 范围       |
|--------|-------------------|------|--------|----------|
| 0x0302 | TemperatureSensor | -    | Simple | Endpoint |

7.7.4.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

7.7.4.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表361。

表361 温度传感器功能集要求

| 标识符    | 名称                     | 客户端/服务端 | 质量等级 | 一致性      |
|--------|------------------------|---------|------|----------|
| 0x0402 | TemperatureMeasurement | Server  | -    | M        |
| 0x0003 | Identify               | Server  | -    | M        |
| 0x0004 | Groups                 | Client  | -    | [Zigbee] |

7.7.4.5 元素限制

此设备类型不覆盖任何cluster规格要求。目前规范还不是特别完善，还有很多设置的功能没有增加，但是基本的cluster已经定义好，此外，规范中还存在很多类似的传感器，基本原理是通过对应的cluster将数据报告到接受方和配置传感器信息，所以下发类似传感器将不解读说明。

7.7.5 湿度传感器

7.7.5.1 概述

湿度传感器（通常为相对湿度传感器）用于报告湿度测量值。

7.7.5.2 类别

以下是湿度传感器的类别信息，详见表362。

表362 湿度传感器类别

| 标识符    | 设备名称           | 上层类别 | 分类     | 范围       |
|--------|----------------|------|--------|----------|
| 0x0307 | HumiditySensor | -    | Simple | Endpoint |

7.7.5.3 条件

请参阅“7.1 基本设备类型”定义以了解一致性标签。

7.7.5.4 功能集要求

每个支持此设备类型的端点应基于下面所定义的一致性来包含这些Cluster。详见下表363。

表363 湿度传感器功能集要求

| 标识符    | 功能集                         | 客户端/服务端 | 质量等级 | 一致性      |
|--------|-----------------------------|---------|------|----------|
| 0x0003 | Identify                    | Server  | -    | M        |
| 0x0405 | RelativeHumidityMeasurement | Server  | -    | M        |
| 0x0004 | Groups                      | Client  | -    | [Zigbee] |

7.7.6 开关传感器

7.7.6.1 概述

开关传感器是一种测量和感应设备，当与可调光灯等照明设备绑定时，能够用于开或关该设备。

7.7.6.2 类别

以下是开关传感器的类别信息，详见表364。

表364 开关传感器类别

| 标识符    | 设备名称         | 上层类别 | 分类     | 范围       |
|--------|--------------|------|--------|----------|
| 0x0850 | On/OffSensor | -    | Simple | Endpoint |

7.7.6.3 条件

请参阅“7.1 基本设备类型”定义以了解一致性标签。

7.7.6.4 功能集要求

支持此设备类型的每个端点都应根据以下定义的符合性来包含这些clusters。详见下表365。

表365 开关传感器功能集要求

| 标识符    | 功能集              | 客户端/服务端 | 质量等级 | 一致性  |
|--------|------------------|---------|------|------|
| 0x0003 | Identify         | Server  | –    | M    |
| 0x0003 | Identify         | Client  | –    | M    |
| 0x0004 | Groups           | Client  | –    | 0    |
| 0x0062 | ScenesManagement | Client  | –    | P, 0 |
| 0x0006 | On/Off           | Client  | –    | M    |
| 0x0008 | LevelControl     | Client  | –    | 0    |
| 0x0300 | ColorControl     | Client  | –    | 0    |

## 7.7.7 烟雾一氧化碳报警器

### 7.7.7.1 概述

烟雾一氧化碳报警器能够感应烟雾或者一氧化碳,或两者都能感应。它能够发出视觉和听觉警报,提示烟雾或一氧化碳的浓度已升高。

烟雾一氧化碳报警器能够进行自我监控,并在出现硬件故障、电池电量严重不足和服务终止时发出视觉和听觉警报。某些声音警报可以选择暂时静音。烟雾一氧化碳报警器能够进行自检,对主传感器进行诊断,并发出循环声响和可视的生命安全警报指示。

部分烟雾报警器可能能够调节灵敏度。烟雾一氧化碳报警器可能具有检测和报告湿度水平、温度水平和污染水平的能力。

### 7.7.7.2 类别

以下是烟雾一氧化碳报警器的类别信息,详见表366。

表366 烟雾一氧化碳报警器类别

| 标识符    | 设备名称         | 上层类别 | 分类     | 范围       |
|--------|--------------|------|--------|----------|
| 0x0076 | SmokeCOAlarm | –    | Simple | Endpoint |

### 7.7.7.3 条件

有关一致性标记,请参阅“7.1 基本设备类型”定义。

### 7.7.7.4 设备类型要求

烟雾一氧化碳报警器设备类型应支持某些endpoint上的PowerSources设备类型实例。详见下表367。

表367 烟雾一氧化碳报警器设备类型要求

| 标识符    | 名称          | 约束条件 | 一致性 |
|--------|-------------|------|-----|
| 0x0011 | PowerSource | min1 | M   |

### 7.7.7.5 功能集要求

支持此设备类型的每个endpoint都应根据以下定义的一致性来包含这些cluster。详见下表368。

表368 烟雾一氧化碳报警器设备功能集要求

| 标识符    | 功能集                                    | 客户端/服务端 | 质量等级 | 一致性 |
|--------|----------------------------------------|---------|------|-----|
| 0x0003 | Identify                               | Server  | –    | M   |
| 0x0004 | Groups                                 | Server  | –    | 0   |
| 0x005C | SmokeCOAlarm                           | Server  | –    | M   |
| 0x0405 | RelativeHumidityMeasurement            | Server  | –    | 0   |
| 0x0402 | TemperatureMeasurement                 | Server  | –    | 0   |
| 0x040C | CarbonMonoxideConcentrationMeasurement | Server  | –    | 0   |

### 7.7.8 空气质量传感器

#### 7.7.8.1 概述

这是空气质量传感器设备类型的一致性定义。  
空气质量传感器是一种用于监测和测量室内或室外环境中与空气质量相关的各种参数的设备。

#### 7.7.8.2 类别

以下是空气质量传感器设备的类别信息，详见表369。

表369 空气质量传感器设备类别

| 标识符    | 设备名称             | 上层类别 | 分类     | 范围       |
|--------|------------------|------|--------|----------|
| 0x002C | AirQualitySensor | -    | Simple | Endpoint |

#### 7.7.8.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

#### 7.7.8.4 功能集要求

基于以下定义的一致性，支持该设备类型的每个端点应包括这些功能集。详见下表370。

表370 空气质量传感器设备功能集要求

| 标识符    | 功能集                                                   | 客户端/服务端 | 质量等级 | 一致性 |
|--------|-------------------------------------------------------|---------|------|-----|
| 0x0003 | Identify                                              | Server  | -    | M   |
| 0x005B | AirQuality                                            | Server  | -    | M   |
| 0x0402 | TemperatureMeasurement                                | Server  | -    | 0   |
| 0x0405 | RelativeHumidityMeasurement                           | Server  | -    | 0   |
| 0x040C | CarbonMonoxideConcentrationMeasurement                | Server  | -    | 0   |
| 0x040D | CarbonDioxideConcentrationMeasurement                 | Server  | -    | 0   |
| 0x0413 | NitrogenDioxideConcentrationMeasurement               | Server  | -    | 0   |
| 0x0415 | OzoneConcentrationMeasurement                         | Server  | -    | 0   |
| 0x042B | FormaldehydeConcentrationMeasurement                  | Server  | -    | 0   |
| 0x042C | PM1 ConcentrationMeasurement                          | Server  | -    | 0   |
| 0x042A | PM2.5 ConcentrationMeasurement                        | Server  | -    | 0   |
| 0x042D | PM10 ConcentrationMeasurement                         | Server  | -    | 0   |
| 0x042F | RadonConcentrationMeasurement                         | Server  | -    | 0   |
| 0x042E | TotalVolatileOrganicCompoundsConcentrationMeasurement | Server  | -    | 0   |

### 7.7.9 水泄漏检测设备类型

#### 7.7.9.1 概述

该章节定义了水泄漏检测设备类型的一致性。

#### 7.7.9.2 类别

以下是水泄漏检测设备的类别信息，详见表371。

表371 水泄漏检测设备类别

| 标识符    | 设备名称              | 上层类别 | 分类     | 范围       |
|--------|-------------------|------|--------|----------|
| 0x0043 | WaterLeakDetector | -    | Simple | Endpoint |

#### 7.7.9.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

#### 7.7.9.4 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表372。

表372 水泄漏检测设备功能集要求

| 标识符    | 名称                        | 客户端/服务端 | 质量等级 | 一致性 |
|--------|---------------------------|---------|------|-----|
| 0x0003 | Identify                  | Server  | —    | M   |
| 0x0045 | BooleanState              | Server  | —    | M   |
| 0x0080 | BooleanStateConfiguration | Server  | —    | O   |

——Identify 功能集

用于标识端点。

——布尔状态功能集

用于指示传感器/探测器的状态。

布尔状态功能集的状态应使用以下方案反映传感器检测的情况。详见下表373。

表373 水泄漏检测设备功能集状态

| 值     | 状态      |
|-------|---------|
| True  | 检测到水泄漏  |
| False | 未检测到水泄漏 |

——布尔状态配置功能集

这用于配置传感器/探测器，并且与布尔状态功能集的配置相关联。

#### 7.7.9.5 元素要求

下表列出了覆盖功能集规范要求的质量和一致性。空白的表格单元格表示该项没有变化，并且应用功能集规范中的值。详见下表374。

表374 元素要求

| 标识符    | 功能集          | 元素    | 名称          | 约束条件 | 通道 | 一致性 |
|--------|--------------|-------|-------------|------|----|-----|
| 0x0045 | BooleanState | Event | StateChange | —    | —  | M   |

### 7.8 暖通空调类设备类型

#### 7.8.1 恒温器

##### 7.8.1.1 概述

恒温器可以是内置的，也可以是独立的温度、湿度或占位传感器，它允许远程或本地设置该设备温度，恒温器能够发送加热和/或冷暖需求给制冷/制热设备（例如，室内空气处理装置）或能够包括一种直接控制加热或冷却装置。

#### 7.8.2 类别

##### 7.8.2.1 概述

暖通空调设备类型的类别如下表375所示。

表375 类别信息

| 标识符    | 设备名称       | 上层类别 | 分类     | 范围       |
|--------|------------|------|--------|----------|
| 0x0301 | Thermostat | —    | Simple | Endpoint |

##### 7.8.2.2 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

##### 7.8.2.3 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表376。

表376 功能集要求

| 标识符    | 名称                                   | 客户端/服务端 | 质量等级 | 一致性    |
|--------|--------------------------------------|---------|------|--------|
| 0x0003 | Identify                             | Server  | –    | M      |
| 0x0004 | Groups                               | Server  | –    | Active |
| 0x0201 | Thermostat                           | Server  | –    | M      |
| 0x0204 | ThermostatUserInterfaceConfiguration | Server  | –    | 0      |
| 0x009B | EnergyPreference                     | Server  | –    | 0      |
| 0x0202 | FanControl                           | Client  | –    | 0      |
| 0x0402 | TemperatureMeasurement               | Client  | –    | 0      |
| 0x0405 | RelativeHumidityMeasurement          | Client  | –    | 0      |
| 0x0406 | OccupancySensing                     | Client  | –    | 0      |

#### 7.8.2.4 元素限制

下面列出了覆盖功能集规范要求的特性和一致性。空白的单元格表示对该项没有更改而是使用功能集规范中的值。详见下表377。

表377 元素限制

| 标识符    | 功能集        | 元素        | 名称                        | 约束条件 | 通道 | 一致性 |
|--------|------------|-----------|---------------------------|------|----|-----|
| 0x0201 | Thermostat | Feature   | ScheduleConfiguration     | –    | –  | X   |
| 0x0201 | Thermostat | Attribute | AlarmMask                 | –    | –  | X   |
| 0x0201 | Thermostat | Command   | GetRelayStatusLog         | –    | –  | X   |
| 0x0201 | Thermostat | Command   | GetRelayStatusLogResponse | –    | –  | X   |

#### 7.8.3 风扇

##### 7.8.3.1 概述

风扇设备通常是独立的，或安装在天花板或墙上，用于在房间内循环空气。

##### 7.8.3.2 类别

以下是风扇的类别信息，详见表378。

表378 风扇类别

| 标识符    | 设备名称 | 上层类别 | 分类     | 范围       |
|--------|------|------|--------|----------|
| 0x002B | Fan  | –    | Simple | Endpoint |

##### 7.8.3.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

##### 7.8.3.4 设备类型要求

风扇可以通过一个或多个额外的设备类型在不同的端点上展示其功能。组合中使用的所有设备都必须遵守系统模型的消歧要求。表中未明确列出的其他设备类型也可以包含在设备组合中，但不被视为设备的核心功能的一部分。详见下表379。

表379 风扇设备类型要求

| 标识符    | 名称         | 约束条件 | 一致性 |
|--------|------------|------|-----|
| 0x0301 | Thermostat | –    | 0   |

##### 7.8.3.5 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些clusters。详见下表380。

表380 功能集要求

| 标识符    | 名称         | 客户端/服务端 | 质量等级 | 一致性 |
|--------|------------|---------|------|-----|
| 0x0003 | Identify   | server  | –    | M   |
| 0x0004 | Groups     | server  | –    | M   |
| 0x0006 | On/Off     | server  | –    | O   |
| 0x0202 | FanControl | server  | –    | M   |

7.8.3.6 功能集限制

——On/OffCluster（Server）说明

虽然FanControlCluster的FanMode属性也包含一个关闭设置,但On/OffCluster是独立于FanMode属性的。

当On/OffCluster的OnOff属性从TRUE转换为FALSE时,如果FanControlCluster的FanMode属性设置为Off以外的值,则稍后当On/OffCluster的OnOff属性从FALSE转换为TRUE时,FanControlCluster的FanMode、SpeedSetting和PercentSetting属性值是可能需要恢复的。

当设备被关闭时,如果FanControlCluster的FanMode属性设置为Off,则上述信息会丢失,因为SpeedSetting和PercentSetting将设置为零。将On/OffCluster与FanControlCluster一起使用,可实现FanMode、SpeedSetting和PercentSetting属性保持不变。在这种情况下,On/OffCluster的OnOff属性将设置为Off,FanControlCluster的SpeedCurrent和PercentCurrent属性设置为零,而不更改FanMode、SpeedSetting和PercentSetting属性。

7.8.3.7 元素限制

下表381列出了覆盖cluster规范要求的特性和一致性。空白的单元格表示对该项没有更改。

表381 元素限制

| 标识符 | 功能集        | 元素        | 名称              | 质量等级 | 约束条件 | 通道s | 一致性    |
|-----|------------|-----------|-----------------|------|------|-----|--------|
| 1   | FanControl | attribute | FanModeSequence | F    | –    | RVO | Matter |

7.8.4 空气净化器

7.8.4.1 概述

空气净化器是一种独立的设备,旨在清洁房间内的空气。

它是一种带有风扇的设备,用于在运行时控制空气速度。可选地,也可以报告其过滤器的状况。

7.8.4.2 类别

以下是空气净化器的类别信息,详见表382。

表382 空气净化器类别

| 标识符    | 设备名称        | 上层类别 | 分类     | 范围       |
|--------|-------------|------|--------|----------|
| 0x002D | AirPurifier | –    | Simple | Endpoint |

7.8.4.3 条件

请参阅“7.1 基本设备类型”定义的一致性标签。

7.8.4.4 设备类型要求

一个空气净化器可以通过不同端点上的一个或多个附加设备类型来展示其功能元素。所有用于组合的设备都应遵守系统模型的消歧要求。其他未在表中明确列出的设备类型也可以包含在设备组合中,但不被视为设备的核心功能的一部分。详见下表383。

表383 设备类型要求

| 标识符    | 名称                | 约束条件 | 一致性 |
|--------|-------------------|------|-----|
| 0x0301 | Thermostat        | —    | 0   |
| 0x0302 | TemperatureSensor | —    | 0   |
| 0x0307 | HumiditySensor    | —    | 0   |
| 0x002C | AirQualitySensor  | —    | 0   |

#### 7.8.4.5 功能集要求

基于以下一致性的定义，支持该设备类型的每个端点应包括这些功能集。详见下表384。

表384 功能集要求

| 标识符    | 功能集                             | 客户端/服务端 | 质量等级 | 一致性 |
|--------|---------------------------------|---------|------|-----|
| 0x0003 | Identify                        | Server  | —    | M   |
| 0x0004 | Groups                          | Server  | —    | 0   |
| 0x0006 | On/Off                          | Server  | —    | 0   |
| 0x0202 | FanControl                      | Server  | —    | M   |
| 0x0071 | HEPAFilterMonitoring            | Server  | —    | 0   |
| 0x0072 | ActivatedCarbonFilterMonitoring | Server  | —    | 0   |

#### 7.8.4.6 功能集限制

——On/OffCluster (Server) 说明

虽然FanControlCluster的FanMode属性也包含一个关闭设置，但On/OffCluster是独立于FanMode属性的。

当On/OffCluster的OnOff属性从TRUE转换为FALSE时，如果FanControlCluster的FanMode属性设置为Off以外的值，则稍后当On/OffCluster的OnOff属性从FALSE转换为TRUE时，FanControlCluster的FanMode、SpeedSetting和PercentSetting属性值是可能需要恢复的。

当设备被关闭时，如果FanControlCluster的FanMode属性设置为Off，则上述信息会丢失，因为SpeedSetting和PercentSetting将设置为零。将On/OffCluster与FanControlCluster一起使用，可实现FanMode、SpeedSetting和PercentSetting属性保持不变。在此情况下，On/OffCluster的OnOff属性将设置为Off，FanControlCluster的SpeedCurrent和PercentCurrent属性设置为零，而不更改FanMode、SpeedSetting和PercentSetting属性。

#### 7.8.4.7 元素要求

不覆盖任何功能集元素。

## 参 考 文 献

- [1] Matter连接标准联盟官方文档- Matter 1.4 中文指导文件
-