

DL/T、Q/GDW

电力行业标准、国家电网公司企业标准

DL/T 698.45-2017、Q/GDW 11778-2017

面向对象的用电信息数据交换协议

Object oriented electric data exchange protocol

泛在电力物联网增补版-征求意见稿

2019-08-27

说明：

1、标黄部分为增补或修订内容；

2、本文件中的协议增补内容并非定稿，在正式发布之前仍可能修改；

3、联系邮箱：1127293259@qq.com；

XXXX - XX - XX 发布

XXXX - XX - XX 实施

国家电网公司

发布

目 次

前言	III
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 符号和缩略语	2
4.1 符号	2
4.2 缩略语	3
5 通信架构	3
5.1 信息交换模型	3
5.2 基于应用连接的数据交换	4
5.3 请求/响应类型的数据交换	4
5.4 通知/确认类型的数据交换	5
5.5 服务器模型	5
6 数据链路层	6
6.1 字节格式	6
6.2 帧格式	6
6.3 传输规则	9
7 应用层	11
7.1 标记规则和编码规则	11
7.2 通用数据类型	11
7.3 数据类型定义	13
7.4 应用层数据单元	26
8 接口类与对象标识	55
8.1 对象模型	55
8.2 接口类	55
8.3 对象标识	97
附录 A (规范性附录) 对象标识定义	110
附录 B (规范性附录) 物理单位枚举定义及数值举例	179
附录 C (规范性附录) 有关一致性协商	182
附录 D (资料性附录) 校验算法	184
附录 E (规范性附录) 安全认证说明	187
附录 F (资料性附录) 安全模式参数	189
附录 G (资料性附录) 状态字、特征字、模式字	192

附录 H（资料性附录） APDU 编码举例	197
编制说明.....	213

前 言

为提高用电信息采集系统的业务适应性、采集效率、安全性和数据溯源性，规范用电信息数据交换协议的通信架构、数据链路层、应用层、接口类与对象标识，制定本标准。

本标准由国家电网公司营销部提出并解释。

本标准由国家电网公司科技部归口。

本标准起草单位：

本标准主要起草人：

本标准首次发布。

本标准在执行过程中的意见或建议反馈至国家电网公司科技部。

面向对象的用电信息数据交换协议

1 范围

本标准规定了用电信息数据交换协议的通信架构、数据链路层、应用层、接口类与对象标识。

本标准适用于用电信息采集系统主站（以下简称“主站”）、采集终端、电能表之间的通信数据交换。水、气、热等表计的数据采集可参照使用。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本文件。凡是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。

GB/T 16262.1—2006 信息技术 抽象语法记法（ASN.1） 第1部分 基本记法规范

GB/T 17966—2000 微处理器系统的二进制浮点运算

CJ/T 188—2004 户用计量仪表数据传输技术条件

DL/T 645—2007 多功能电能表通信协议

DL/T 790.6—2010 采用配电线载波系统的配电自动化 第6部分 A-XDR编码规则

GB/T 16263.2—2006 信息技术ASN.1编码规则 第2部分：紧缩编码规则（PER）规范

3 术语和定义

下列术语和定义适用于本文件。

3.1

面向对象的用电信息数据交换协议 `object oriented electric data exchange protocol`

基于面向对象建模方法建立的一套适用于用电信息采集系统的互操作性数据交换通信协议。面向对象建模以接口类实现继承关系，以对象来封装数据及操作，以对象为互操作的基本要素。

3.2

对象标识 `object identification`

标识终端中对象唯一名称的编码。

3.3

逻辑名 `logical name`

用于标识接口类的实例，它是接口类的第一个属性，它的值与对象标识一致。

3.4

类标识码 `class_id`

用于区别对象接口类的标识码，即接口类的名称。

3.5

服务器地址 server address

客户机/服务器（Client/Server）访问模型中的服务器（Server）的通信地址。

3.6

逻辑地址 logic address

服务器模型中逻辑设备的地址。

3.7

客户机地址 client address

客户机/服务器（Client/Server）访问模型中的客户机（Client）的通信地址。

注：主站访问采集终端时，采集终端为服务器，主站为客户机；主站访问电能表时，电能表为服务器，主站为客户机；采集终端访问电能表时，电能表为服务器，采集终端为客户机。

3.8

采集启动时标 acquisition start time

启动采集任务时的设备时钟当前值，其值只与启动时刻有关，与执行的时间长短无关。

3.9

采集成功时标 acquisition time

客户机成功接收到服务器响应时的设备时钟当前值。

3.10

采集存储时标 acquisition storage time

采集到的数据进行存储的时间。

3.11

组地址 group address

具有某一相同属性的设备群组编码，如属于同一行业，同一变电站，同一线路，可以响应同一个命令。

3.12

通配地址 the wildcard address

在十进制编码表示的地址码中出现一位或多位采用了通配符的地址码。

3.13

消息鉴别码 Message Authentication Code

用于鉴别消息的完整性的固定长度的认证标识。

4 符号和缩略语

4.1 符号

下列符号适用于本文件。

::=: 定义为

4.2 缩略语

下列缩略语适用于本文件。

A: 地址域 (Address)
 ACD: 请求访问标识 (Ask Call Demand)
 AD: 采集数据 (Acquired Data)
 APDU: 应用层数据单元 (Application layer Data Unit)
 A-XDR: 抽象的外部数据表示 (Abstract eXternal Data Representation)
 B: 二进制 (Binary)
 CA: 客户机通信地址 (Client Address)
 CSD: 列选择描述符 (Column Selection Descriptor)
 DAR: 数据访问结果 (Data Access Result)
 DIR: 传输方向位 (Direction)
 ESAM: 嵌入式安全控制模块 (Embedded Secure Access Module)
 FCS: 帧校验 (Frame Check Sum)
 H: 十六进制 (Hex)
 HCS: 帧头校验 (Head Check Sum)
 IC: 接口类 (Interface Class)
 LSB: 最低有效位 (Least Significant bit)
 MAC: 消息鉴别码 (Message Authentication Code)
 MS: 表计集合 (Meter Set)
 OAD: 对象属性描述符 (Object Attribute Descriptor)
 OI: 对象标识 (Object Identify)
 OMD: 对象方法描述符 (Object Method Descriptor)
 PIID: 序号及优先标志 (Priority and Invoke ID)
 PIID-ACD: 带请求访问标识的序号及优先标志 (Priority and Invoke ID with ACD)
 PRM: 启动标识位 (Primary Request Message)
 RCSD: 记录列选择描述符 (Record Column Selection Descriptor)
 RN: 随机数 (Random Numbers)
 ROAD: 记录型对象属性描述符 (Record Object Attribute Descriptor)
 RSD: 记录选择描述符 (Record Selection Descriptor)
 SA: 服务器通信地址 (Server Address)
 SC: 扰码 (Scrambling Code)
 TI: 时间间隔 (Time Interval)
 TSA: 目标服务器地址 (Target Server Address)

5 通信架构

5.1 信息交换模型

本标准信息交换模型见图1。

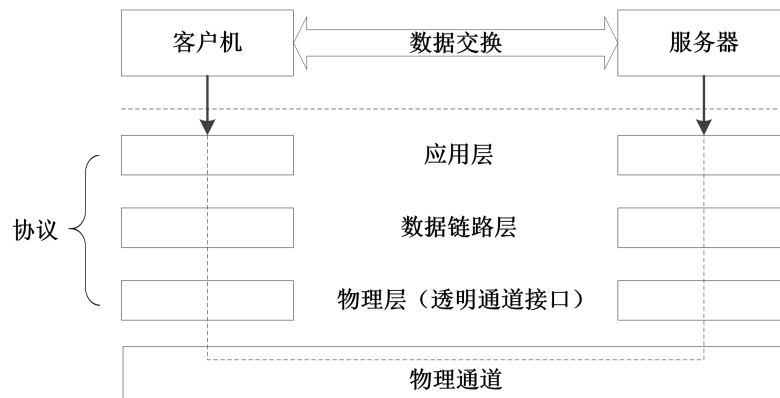


图1 信息交换模型

客户机和服务器之间的信息交换借助于通信协议实现。客户机和服务器的应用进程使用协议应用层的服务，应用层是唯一包含服务组件的协议层，应用层数据单元（APDU）通过数据链路层协议传输帧的链路用户数据域传输。

5.2 基于应用连接的数据交换

本标准中服务器和客户机之间是基于应用连接的数据交换，见图2。

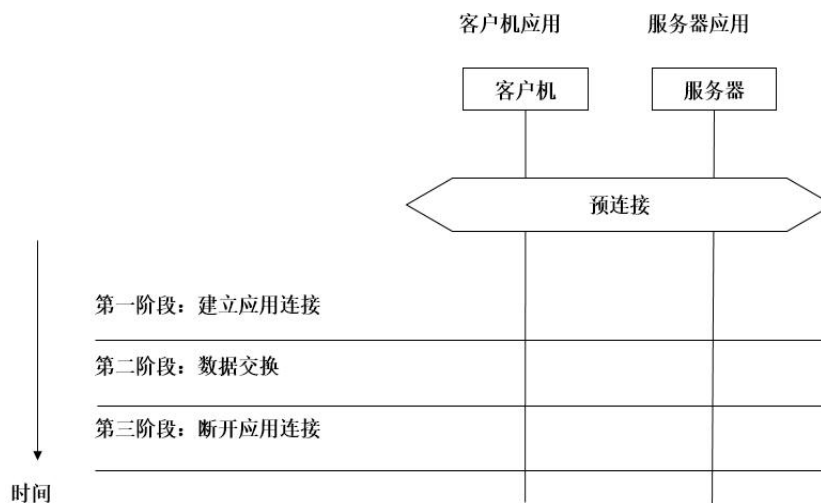


图2 基于应用连接的数据交换

客户机和服务器在开始通信前，通信信道必须先完成预连接。预连接建立后，默认具有一个最低权限的应用连接，客户机和服务器之间可直接进行数据交换。当客户机需要得到更高权限的服务器服务时，客户机必须发起建立更高权限的应用连接。

5.3 请求/响应类型的数据交换

本标准支持客户机应用进程向服务器应用进程提出服务请求，服务器应用进程向客户机应用进程提供远程服务响应，见图3。

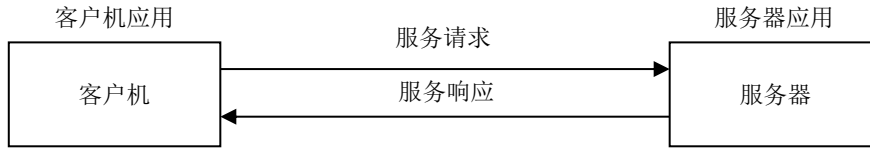


图3 请求/响应类型数据交换

5.4 通知/确认类型的数据交换

本标准支持服务器应用进程根据客户机预先定制的主动上报内容，向客户机应用进程提供远程主动上报数据服务，客户机应用进程向服务器应用进程回复服务确认，见图4。

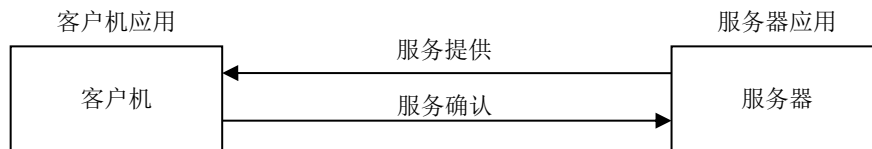


图4 通知/确认类型数据交换

5.5 服务器模型

服务器模型见图5。

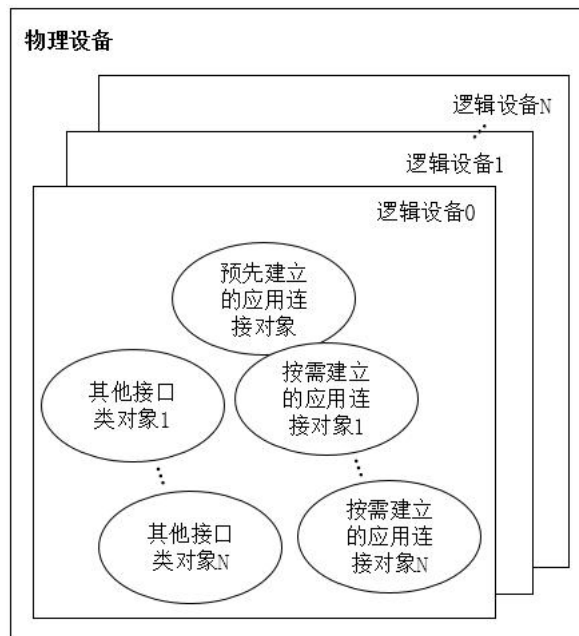


图5 服务器模型

服务器模型要求如下：

- a) 物理设备由若干逻辑设备构成，逻辑设备0为必须具备的逻辑设备；

- b) 每个逻辑设备由若干可访问的接口类对象构成，包括一个预先建立的应用连接对象、若干个按需建立的应用连接对象、若干个其他接口类对象；
- c) 预先建立的应用连接对象为逻辑设备必须具备的对象；
- d) 按需建立的应用连接对象为可选对象，根据所需的访问权限进行配置；
- e) 其他接口类对象为可选对象，根据设备功能所需进行配置。

6 数据链路层

6.1 字节格式

链路层帧的基本单元为8位字节，传输顺序为低位在前，高位在后；低字节在前，高字节在后。

6.2 帧格式

6.2.1 帧结构

本标准采用的帧格式见图6。

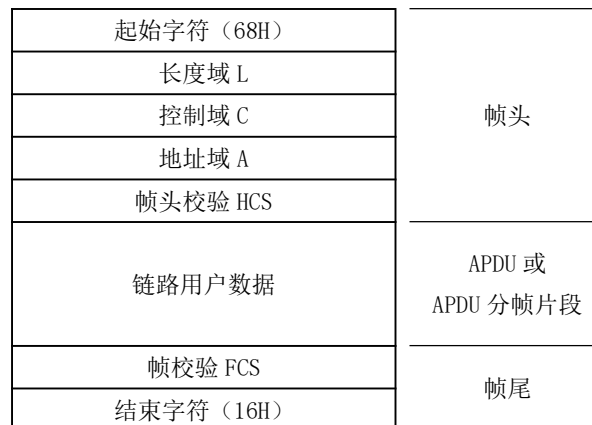


图 6 帧格式

6.2.2 长度域 L

长度域L由2字节组成，采用BIN编码，定义见图7。

bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
保留	帧数据长度单位	帧数据长度值													

图 7 长度域 L 定义

其中：

- a) bit0...bit13: 帧数据长度值，是传输帧中不包含起始字符和结束字符的数据长度；
- b) bit14: 帧数据长度单位，0 表示帧数据长度单位为字节，1 表示帧数据长度单位为千字节；
- c) bit15: 保留。

6.2.3 控制域 C

6.2.3.1 控制域 C 定义

控制域C为1个字节，按位或位的组合使用，定义见图8。

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
传输方向 DIR	启动标志 PRM	分帧标志	保留	扰码标志 SC	功能码		

图8 控制域C定义

6.2.3.2 传输方向及启动标志位

传输方向位及启动标志位定义：

- a) 传输方向位：bit7=0 表示此帧是由客户机发出的；bit7=1 表示此帧是由服务器发出的；
 - b) 启动标志位：bit6=0 表示此帧是由服务器发起的；bit6=1 表示此帧是由客户机发起的。
- 传输方向位DIR和启动标志位PRM组合意义见表1。

表1 传输方向位 DIR 和启动标志位 PRM 组合意义

DIR	PRM	组合意义
0	0	客户机对服务器上报的响应
0	1	客户机发起的请求
1	0	服务器发起的上报
1	1	服务器对客户机请求的响应

6.2.3.3 分帧标志位

分帧标志位：bit5=0，表示此帧链路用户数据为完整APDU；bit5=1，表示此帧链路用户数据为APDU片段。

6.2.3.4 扰码标志位

扰码标志位：bit3=0，表示此帧链路用户数据不加扰码；bit3=1，表示此帧链路用户数据加扰码，发送时链路用户数据按字节加33H。

6.2.3.5 功能码

功能码采用BIN编码，定义见表2。

表2 功能码定义

功能码	服务类型	应用说明
0	保留	
1	链路管理	链路连接管理（登录，心跳，退出登录）
2	保留	
3	用户数据	应用连接管理及数据交换服务
4...7	保留	

6.2.4 地址域 A

6.2.4.1 地址域组成

地址域A由可变字节数的服务器地址SA和1字节的客户机地址CA组成，定义见图9。

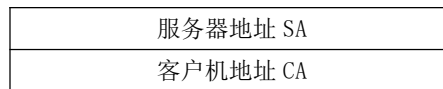


图9 地址域定义

6.2.4.2 服务器地址 SA

6.2.4.2.1 服务器地址定义

服务器地址SA由1字节地址特征和N个字节地址组成。定义见图10。

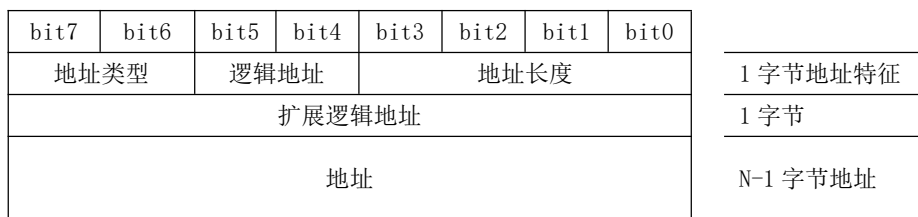


图10 服务器地址 SA 定义

地址特征定义：

- bit0...bit3：为地址的字节数，取值范围：0...15，对应表示1...16个字节长度；
- bit4...bit5：逻辑地址；
bit5=0表示无扩展逻辑地址，bit4取值0和1分别表示逻辑地址0和1；
bit5=1表示有扩展逻辑地址，bit4备用；地址长度N包含1个字节的扩展逻辑地址，取值范围2...255，表示逻辑地址2...255。
- bit6...bit7：为服务器地址的地址类型，0表示单地址，1表示通配地址，2表示组地址，3表示广播地址。

扩展逻辑地址和地址要求如下：

- 扩展逻辑地址取值范围2...255；
- 编码方式为压缩BCD码，0保留；
- 当服务器地址的十进制位数为奇数时，最后字节的bit3...bit0用FH表示。

6.2.4.2.2 单地址

单地址的长度为可变字节。

6.2.4.2.3 组地址

组地址的长度为可变字节。组地址对系统中属于该群组的服务器有效，无需应答。

6.2.4.2.4 通配地址

通配地址的长度为可变字节。每字节二进制高低各4位分别编码表示两个0到9的十进制数或通配符AH。通配符按十进制位使用，对应的十进制位为AH时，表示该十进制位可为0到9的任意值。

6.2.4.2.5 广播地址

广播地址的长度固定为1字节，广播地址=AAH。广播地址对系统所有服务器有效，无需应答。

6.2.4.3 客户机地址 CA

客户机地址CA用1字节表示，0表示不关注客户机地址。

6.2.5 帧头校验 HCS

帧头校验HCS为2字节，是对帧头部分不包含起始字符和HCS本身的所有字节的校验，校验算法见附录 D。

6.2.6 链路用户数据

链路用户数据包含一个APDU或APDU分帧片段，APDU（应用层数据单元）见7。

6.2.7 帧校验 FCS

帧校验FCS为2字节，是对整帧不包含起始字符、结束字符和FCS本身的所有字节的校验，校验算法见附录 D。

6.3 传输规则

6.3.1 字节规则

采用串行通信方式发送数据时：

- a) 线路空闲状态为二进制 1；
- b) 在有效数据帧前加 4 个 FEH 作为前导码；
- c) 数据链路层帧的字节之间无线路空闲间隔；两帧之间的线路空闲间隔至少 33 位。

6.3.2 分帧传输

6.3.2.1 分帧传输规则

当一个APDU数据组帧的长度超过协商的最大帧长度时，采用分帧传输。采用分帧传输时，控制域中分帧标志位置1。数据接收端应对分帧进行逐条确认。

6.3.2.2 分帧传输格式定义

分帧传输时，数据链路层的链路用户数据为分帧传输帧，分为数据帧和确认帧；分帧传输的数据帧包含分帧格式域和APDU片段，格式定义见图11。分帧传输的确认帧仅包含分帧格式域，不含APDU片段。

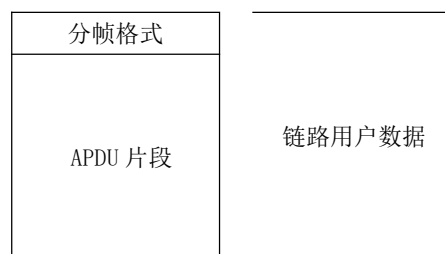


图 11 分帧传输帧格式定义

分帧格式域为2字节，按位或位的组合使用，定义见图12。

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
bit15...bit14		bit13	bit12	bit11	bit10	bit9	bit8

图 12 分帧格式域定义

位定义如下：

- bit0...bit11：表示分帧传输的帧序号，取值范围 0...4095，循环使用；
- bit12...bit13：保留；
- bit14...bit15：表示分帧类型，定义见表 3。

表 3 分帧类型定义

分帧类型	意义
0	表示分帧传输起始帧
1	表示分帧传输末尾帧
2	表示分帧传输确认帧
3	表示分帧传输中间帧

6.3.2.3 分帧传输交互规则

分帧传输交互可由服务器或客户机任意一侧发起，适用于主动发起的数据服务或者被动应答的数据服务，发送方将APDU应用数据单元分割成若干个片段，通过分帧方式依次发送。

分帧传输交互过程如下：

- 由服务器或客户机启动的数据分帧传输过程见图 13。

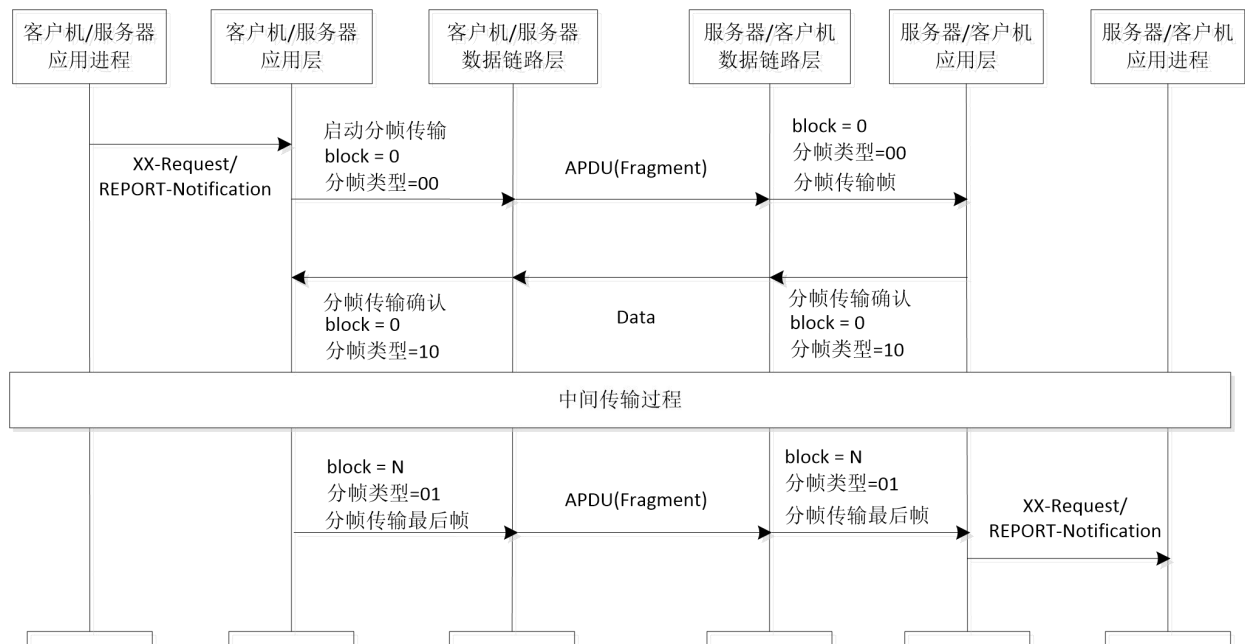


图 13 服务器/客户机启动传输的数据请求服务的分帧过程

- 服务器响应客户机请求的数据分帧传输过程见图 14。

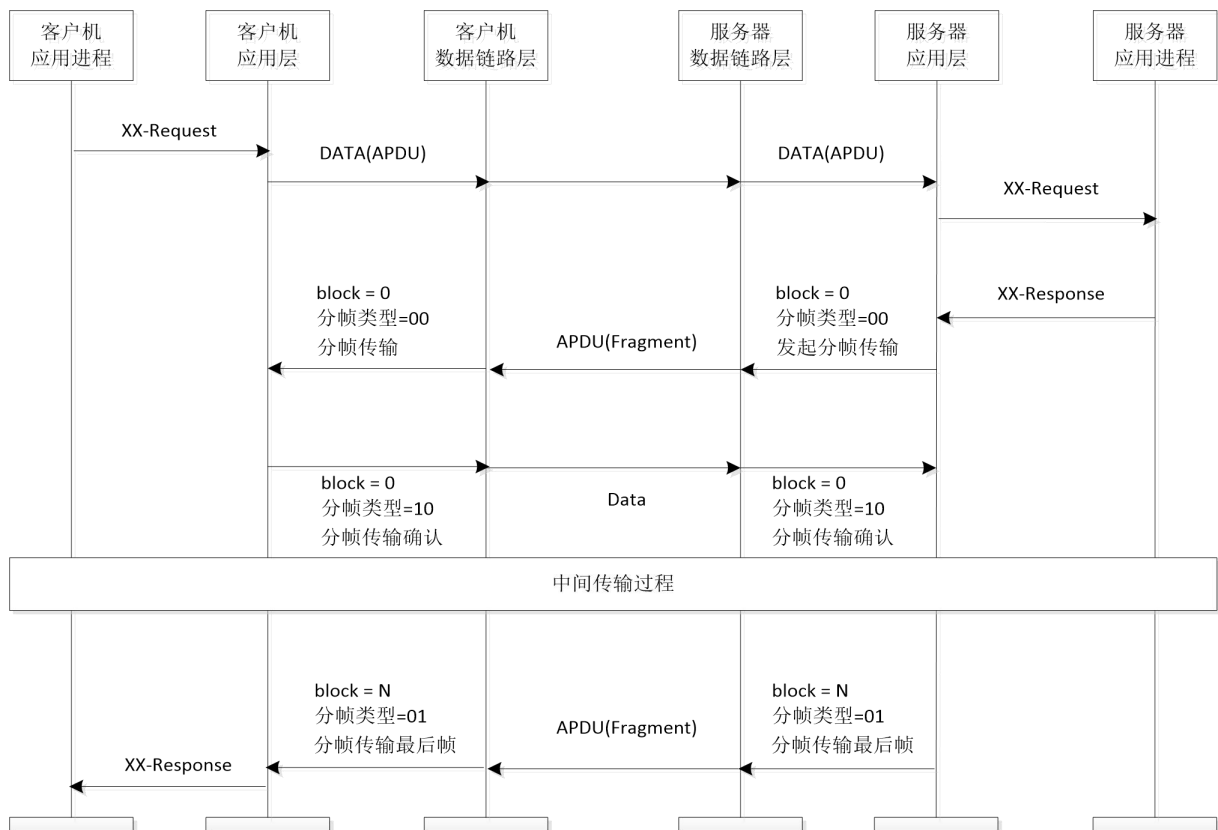


图 14 服务器响应数据请求的服务分帧过程

分帧过程如下：

- 分帧的第一帧数据，帧序号 (block) = 0，分帧类型=0；
- 分帧传输确认，分帧类型=2，block 值为最近一次收到正确的帧序号；
- 分帧的发送方在接收到确认帧后传输下一个数据帧，分帧类型=3，分帧序号 block=接收到确认帧的帧序号+1，重复 b) 和 c)；
- 分帧的最后一帧，block=最后一帧的帧序号，分帧类型=1，发送方在接收到最后一个确认帧后，分帧传输过程完成。

7 应用层

7.1 标记规则和编码规则

应用层数据单元（APDU）的标记规则遵循ASN.1的抽象语法，详见GB/T 16262.1—2006。应用层数据单元（APDU）的编码规则遵循A-XDR，详见DL/T 790.6—2010。

应用层紧凑服务的数据单元（APDU）的编码规则遵循PER，详见GB/T 16263.2—2006。

7.2 通用数据类型

接口类及对象实例使用的数据类型见表4。

表4 接口类及对象实例使用的数据类型定义

类型描述	标记	定义	数值范围
NULL	0	空	
array	1	SEQUENCE OF Data（见7.3.1） 数组的元素在对象属性或方法的描述中定义	
structure	2	SEQUENCE OF Data（见7.3.1） 结构的元素在对象属性或方法的描述中定义	
bool	3	布尔值	1 或 0
bit-string	4	位串	
double-long	5	32 位整数	$-2^{31} \dots 2^{31}-1$
double-long-unsigned	6	32 位正整数	$0 \dots 2^{32}-1$
保留	7-8		
octet-string	9	8 位字节串	
visible-string	10	ASCII 字符串	
保留	11		
UTF8-string	12	UTF-8 编码的字符串	
保留	13-14		
integer	15	8 位整数	-128...127
long	16	16 位整数	-32768...32767
unsigned	17	8 位正整数	0...255
long-unsigned	18	16 位正整数	0...65535
保留	19		
long64	20	64 位整数	$-2^{63} \dots 2^{63}-1$
long64-unsigned	21	64 位正整数	$0 \dots 2^{64}-1$
enum	22	枚举的元素在对象属性或方法的描述中定义	0...255
float32	23	32 位浮点数	
float64	24	64 位浮点数	
date_time	25	octet-string (SIZE (10))	
date	26	octet-string (SIZE (5))	
time	27	octet-string (SIZE (3))	
date_time_s	28	octet-string (SIZE (7))	
保留	29-79		
OI	80	见 7.3.5	
OAD	81	见 7.3.6	
ROAD	82	见 7.3.7	

类型描述	标记	定义	数值范围
OMD	83	见 7.3.8	
TI	84	见 7.3.9	
TSA	85	见 7.3.10	
MAC	86	见 7.3.11	
RN	87	见 7.3.12	
Region	88	见 7.3.13	
Scaler_Unit	89	见 7.3.14	
RSD	90	见 0	
CSD	91	见 0	
MS	92	见 7.3.17	
SID	93	见 7.3.18	
SID_MAC	94	见 7.3.19	
COMDCB	95	见 7.3.20	
RCSID	96	见 7.3.21	
保留	97-255		

注：表中的“位”表示1个二进制信息单位（bit）。

7.3 数据类型定义

7.3.1 Data 数据类型

Data的数据类型定义见表5。

表5 Data 数据类型定义

数据类型定义	说明
Data ::= CHOICE { NULL [0], array [1], structure [2], bool [3], bit-string [4], double-long [5], double-long-unsigned [6], octet-string [9], visible-string [10], UTF8-string [12], integer [15], long [16], unsigned [17], long-unsigned [18], long64 [20], long64-unsigned [21], enum [22],	见 7.2。

数据类型定义	说明
float32 [23],	
float64 [24],	
date_time [25],	
date [26],	
time [27],	
date_time_s [28],	
OI [80],	
OAD [81],	
ROAD [82],	
OMD [83],	
TI [84],	
TSA [85],	
MAC [86],	
RN [87],	
Region [88],	
Scaler_Unit [89],	
RSD [90],	
CSD [91],	
MS [92],	
SID [93],	
SID_MAC [94],	
COMDCB [95],	
RCSID [96]	
}	

7.3.2 浮点数据类型

浮点数据类型包括float32以及float64，格式定义见GB/T 17966—2000。

7.3.3 PIID 数据类型

服务序号及优先标志PIID (Priority and Invoke ID) 的数据类型定义见表6。

表6 PIID 数据类型定义

数据类型定义	说明
PIID ::= unsigned	PIID用于客户机APDU (Client-APDU) 的服务数据类型中，其中： bit7 (服务优先级) —— 0: 普通优先级，1: 高优先级，在应答APDU中，其值与请求的APDU相同。 bit6 (保留)。 bit0...bit5 (服务序号) —— 二进制编码表示0...63，在应答APDU中，其值与请求的APDU相同。

7.3.4 PIID-ACD 数据类型

带ACD标志位的服务序号及优先标志PIID-ACD (Priority and Invoke ID with ACD) 的数据类型定义见表7。

表 7 PIID-ACD 数据类型定义

数据类型定义	说明
PIID-ACD ::= unsigned	PIID-ACD 用于服务器 APDU (Server-APDU) 的服务数据类型中, 其中: bit7 (服务优先级) —— 见表 6。 bit6 (请求访问 ACD) —— 0: 不请求, 1: 请求。 bit0...bit5 (服务序号) —— 见表 6。

7.3.5 OI 数据类型

对象标识数据类型OI定义见表8。

表 8 OI 数据类型定义

数据类型定义	说明
OI ::= long-unsigned	

7.3.6 OAD 数据类型

对象属性描述符OAD (Object Attribute Descriptor) 的数据类型定义见表9。

表 9 OAD 数据类型定义

数据类型定义	说明
OAD ::= SEQUENCE { 对象标识 OI, 属性标识及其特征 unsigned, 属性内元素索引 unsigned (1...255) }	对象标识——见 8.2.29。 属性标识及其特征, 其中: 1) bit0...bit4 编码表示对象属性编号, 取值 0...31, 其中 0 表示整个对象属性, 即对象的所有属性, 见 8.2.1; 2) bit5...bit7 编码表示属性特征, 属性特征是对象同一个属性在不同快照环境下取值模式, 取值 0...7, 特征含义在具体类属性中描述。 属性内元素索引——00H 表示整个属性全部内容。如果属性是结构或数组, 01H 指向对象属性的第一个元素; 如果属性是记录型的存储区, 非 0 值 n 表示最近第 n 次的记录。

7.3.7 ROAD 数据类型

记录型对象属性描述符ROAD (Record Object Attribute Descriptor) 的数据类型定义见表10。

表 10 ROAD 数据类型定义

数据类型定义	说明
ROAD ::= SEQUENCE {	ROAD 用于描述记录型对象中的一个或若干个关联对象属性。 OAD——见 7.3.6。

数据类型定义	说明
对象属性描述符 OAD, 关联对象属性描述符 SEQUENCE OF OAD }	

7.3.8 OMD 数据类型

对象方法描述符OMD (Object Method Descriptor) 的数据类型定义见表11 。

表 11 OMD 数据类型定义

数据类型定义	说明
OMD ::= SEQUENCE { 对象标识 OI, 方法标识 unsigned (1...255), 操作模式 unsigned (0) }	OMD 用于描述对象的方法。 对象标识——见 8.2.29 。 方法标识——对象方法编号，见 8.2.1 。 操作模式——值默认为 0。

7.3.9 TI 数据类型

时间间隔TI (Time Interval) 的数据类型定义见表12 。

表 12 TI 数据类型定义

数据类型定义	说明
TI ::= SEQUENCE { 单位 ENUMERATED { 秒 (0), 分 (1), 时 (2), 日 (3), 月 (4), 年 (5) }, 间隔值 long-unsigned }	TI 用于表示时间间隔的间隔值及时间单位，间隔值为 0 表示无间隔。

7.3.10 TSA 数据类型

目标服务器地址TSA (Target Server Address) 的数据类型定义见表13 。

表 13 TSA 数据类型定义

数据类型定义	说明
TSA ::= octet-string (SIZE (2...17))	见 6.2.4.2.1 。

7.3.11 MAC 数据类型

数据安全MAC (Message Authentication Code) 的数据类型定义见表14。

表 14 MAC 数据类型定义

数据类型定义	说明
MAC ::= octet-string	见 3.13。

7.3.12 RN 数据类型

随机数RN (Random Number) 的数据类型定义见表15。

表 15 RN 数据类型定义

数据类型定义	说明
RN ::= octet-string	ESAM 生成的用于加密的信息串。

7.3.13 Region 区间类型

区间类型Region的数据类型定义见表16。

表 16 Region 数据类型定义

数据类型定义	说明
<pre>Region ::= SEQUENCE { 单位 ENUMERATED { 前闭后开 (0), 前开后闭 (1), 前闭后闭 (2), 前开后开 (3) }, 起始值 Data, 结束值 Data }</pre>	Region 用于描述数据的区间范围，包括以下四种：(起始值，结束值)、[起始值，结束值)、(起始值，结束值]、[起始值，结束值]。

7.3.14 Scaler_Unit 数据类型

换算及单位Scaler_Unit的数据类型定义见表17。

表 17 Scaler_Unit 数据类型定义

数据类型定义	说明
<pre>Scaler_Unit ::= SEQUENCE { 换算 integer, 单位 ENUMERATED }</pre>	<p>换算——倍数因子的指数，基数为 10；如数值不是数字的，则换算应被置 0。</p> <p>单位——枚举类型定义物理单位，见附录 B。</p>

数据类型定义	说明
}	

7.3.15 RSD 数据类型

记录行选择描述符RSD (Record Selection Descriptor) 的数据类型定义见表18。

表 18 RSD 数据类型定义

数据类型定义	说明
<pre>RSD ::= CHOICE { 不选择 [0] NULL, 选择方法 1 [1] Selector1, 选择方法 2 [2] Selector2, 选择方法 3 [3] Selector3, 选择方法 4 [4] Selector4, 选择方法 5 [5] Selector5, 选择方法 6 [6] Selector6, 选择方法 7 [7] Selector7, 选择方法 8 [8] Selector8, 选择方法 9 [9] Selector9, 选择方法 10 [10] Selector10 }</pre>	RSD 用于选择记录型对象属性的记录，即二维记录表的行选择，其通过对构成记录的对象属性数值进行选择，范围选择区间：前闭后开，即[起始值，结束值)。
<pre>Selector1 ::= SEQUENCE { 对象属性描述符 OAD, 数值 Data }</pre>	Selector1 为选择对象的指定值。
<pre>Selector2 ::= SEQUENCE { 对象属性描述符 OAD, 起始值 Data, 结束值 Data, 数据间隔 Data }</pre>	Selector2 为选择对象区间内连续间隔值。 数据间隔——是和 OAD 相关的类型，=NULL 表示无间隔（即：指定区间内全部）。 当对象属性描述符的数值类型为 date_time 或 date_time_s 类型时，其间隔的数据类型为 TI；其他类型下通常间隔数据类型与数值类型相同。
<pre>Selector3 ::= SEQUENCE OF Selector2</pre>	Selector3 为多个选择对象区间内连续间隔值的并集。
<pre>Selector4 ::= SEQUENCE { 采集启动时间 date_time_s, 表计集合 MS }</pre>	Selector4 为指定表计集合、指定采集启动时间。 MS——见 7.3.17。
<pre>Selector5 ::= SEQUENCE</pre>	Selector5 为指定表计集合、指定采集存储时间。

数据类型定义	说明
{ 采集存储时间 date_time_s, 表计集合 MS }	MS——见 7.3.17 。
Selector6::=SEQUENCE { 采集启动时间起始值 date_time_s, 采集启动时间结束值 date_time_s, 时间间隔 TI, 表计集合 MS }	Selector6 为指定表计集合、指定采集启动时间区间内连续间隔值。 MS——见 7.3.17 。
Selector7::=SEQUENCE { 采集存储时间起始值 date_time_s, 采集存储时间结束值 date_time_s, 时间间隔 TI, 表计集合 MS }	Selector7 为指定表计集合、指定采集存储时间区间内连续间隔值。 MS——见 7.3.17 。
Selector8::=SEQUENCE { 采集成功时间起始值 date_time_s, 采集成功时间结束值 date_time_s, 时间间隔 TI, 表计集合 MS }	Selector8 为指定表计集合、指定采集成功时间区间内连续间隔值。 MS——见 7.3.17 。
Selector9::=SEQUENCE { 上第 n 次记录 unsigned }	Selector9 为指定选取上第 n 次记录。
Selector10::=SEQUENCE { 上 n 条记录 unsigned, 表计集合 MS }	Select10 为指定选取最新的 n 条记录。 MS——见 7.3.17 。

7.3.16 CSD 数据类型

列选择描述符CSD（Column Selection Descriptor）的数据类型定义见表19 。

表 19 CSD 数据类型定义

数据类型定义	说明
CSD ::= CHOICE { 对象属性描述符 [0] OAD, 记录型对象属性描述符 [1] ROAD }	CSD 用于描述记录型对象中记录的列关联对象属性。 OAD——见 7.3.6。 ROAD——见 7.3.7。

7.3.17 MS 数据类型

表计集合MS (Meter Set) 的数据类型定义见表20。

表 20 MS 数据类型定义

数据类型定义	说明
MS ::= CHOICE { 无表计 [0], 全部用户地址 [1], 一组用户类型 [2] SEQUENCE OF unsigned, 一组用户地址 [3] SEQUENCE OF TSA, 一组配置序号 [4] SEQUENCE OF long-unsigned, 一组用户类型区间 [5] SEQUENCE OF Region, 一组用户地址区间 [6] SEQUENCE OF Region, 一组配置序号区间 [7] SEQUENCE OF Region }	无表计——相当于无效配置。 全部用户地址——全部可采集的表计。 一组用户类型——指定的若干用户类型的表计。 一组用户地址——指定的若干表计通信地址的表计。 一组配置序号——指定的若干表计配置序号的表计。 一组用户类型区间——按数组形式给出用户类型范围。 一组用户地址区间——按数组形式给出表计通信地址范围。 一组配置序号区间——按数组形式给出表计档案配置序号范围。

7.3.18 SID 安全标识类型

安全标识SID (Security Identifier) 的数据类型定义见表21。

表 21 SID 数据类型定义

数据类型定义	说明
SID ::= SEQUENCE { 标识 double-long-unsigned, 附加数据 octet-string }	ESAM 所属安全标识。

7.3.19 SID_MAC 标识类型

SID_MAC的数据类型定义见表22。

表 22 SID_MAC 数据类型定义

数据类型定义	说明
SID_MAC ::= SEQUENCE {	ESAM 所属安全标识以及消息鉴别码。

数据类型定义	说明
安全标识 SID, 消息鉴别码 MAC }	

7.3.20 COMDCB 数据类型

串口控制块COMDCB的数据类型定义见表23。

表 23 COMDCB 数据类型定义

数据类型定义	说明
COMDCB ::= SEQUENCE { 波特率 ENUMERATED { 300bps (0), 600bps (1), 1200bps (2), 2400bps (3), 4800bps (4), 7200bps (5), 9600bps (6), 19200bps (7), 38400bps (8), 57600bps (9), 115200bps (10), 自适应 (255) }, 校验位 ENUMERATED {无校验 (0), 奇校验 (1), 偶校验 (2)}, 数据位 ENUMERATED {5 (5), 6 (6), 7 (7), 8 (8)}, 停止位 ENUMERATED {1 (1), 2 (2)}, 流控 ENUMERATED {无 (0), 硬件 (1), 软件 (2)} }	

7.3.21 RCSD 数据类型

记录列选择描述符RCSD (Record Column Selection Descriptor) 的数据类型定义见表24。

表 24 RCSD 数据类型定义

数据类型定义	说明
RCSD ::= SEQUENCE OF CSD	RCSD 用于选择记录型对象属性中记录的某列或某几列内容，即二维记录表的列选择。 当无 OAD 时，RCSD=0，即 SEQUENCE OF 的数据项个数为 0，表示全选。

7.3.22 DAR 数据类型

数据访问结果DAR (Data Access Result) 的数据类型定义见表25。

表 25 DAR 数据类型定义

数据类型定义	说明
DAR ::= ENUMERATED { 成功 (0),	DAR 采用枚举方式来描述数据访问的各种可能结果。

数据类型定义	说明
硬件失效 (1),	
暂时失效 (2),	
拒绝读写 (3),	
对象未定义 (4),	
对象接口类不符合 (5),	
对象不存在 (6),	
类型不匹配 (7),	
越界 (8),	
数据块不可用 (9),	
分帧传输已取消 (10),	
不处于分帧传输状态 (11),	
块写取消 (12),	
不存在块写状态 (13),	
数据块序号无效 (14),	
密码错/未授权 (15),	
通信速率不能更改 (16),	
年时区数超 (17),	
日时段数超 (18),	
费率数超 (19),	
安全认证不匹配 (20),	
重复充值 (21),	
ESAM 验证失败 (22),	
安全认证失败 (23),	
客户编号不匹配 (24),	
充值次数错误 (25),	
购电超囤积 (26),	
地址异常 (27),	
对称解密错误 (28),	
非对称解密错误 (29),	
签名错误 (30),	
电能表挂起 (31),	
时间标签无效 (32),	
请求超时 (33),	
ESAM 的 PIP2 不正确 (34),	
ESAM 的 LC 错误 (35),	
其它 (255)	
}	

7.3.23 ConnectMechanismInfo 数据类型

应用连接请求认证的机制信息ConnectMechanismInfo的数据类型定义见表26。

表 26 ConnectMechanismInfo 数据类型定义

数据类型定义	说明
ConnectMechanismInfo ::= CHOICE { 公共连接 [0] NullSecurity, 一般密码 [1] PasswordSecurity, 对称加密 [2] SymmetrySecurity, 数字签名 [3] SignatureSecurity }	建立应用连接的机制信息。
NullSecurity ::= NULL	
PasswordSecurity ::= visible-string	
SymmetrySecurity ::= SEQUENCE { 密文 1 octet-string, 客户机签名 1 octet-string }	密文 1 为对客户机产生的随机数加密得到的密文。
SignatureSecurity ::= SEQUENCE { 密文 2 octet-string, 客户机签名 2 octet-string }	密文 2 为客户机对服务器产生的主站证书等数据加密信息。 客户机签名 2 为客户机对密文 2 的签名。

7.3.24 ConnectResult 数据类型

应用连接请求认证的结果ConnectResult的数据类型定义见表27。

表 27 ConnectResult 数据类型定义

数据类型定义	说明
ConnectResult ::= ENUMERATED { 允许建立应用连接 (0), 密码错误 (1), 对称解密错误 (2), 非对称解密错误 (3), 签名错误 (4), 协议版本不匹配 (5), 其他错误 (255) }	ConnectResult 采用枚举方式表示应用连接请求认证的结果。

7.3.25 ConnectResponseInfo 数据类型

应用连接请求的认证响应信息ConnectResponseInfo的数据类型定义见表28。

表 28 ConnectResponseInfo 数据类型定义

数据类型定义	说明
ConnectResponseInfo ::= SEQUENCE { 认证结果 ConnectResult, 认证附加信息 SecurityData OPTIONAL } SecurityData ::= SEQUENCE { 服务器随机数 RN, 服务器签名信息 octet-string }	ConnectResponseInfo 用于表示应用连接请求的认证响应信息。

7.3.26 date_time 数据类型

日期时间date_time的数据类型定义见表29。

表 29 date_time 数据类型定义

数据类型定义	说明
date_time ::= SEQUENCE { year long-unsigned, month unsigned, day_of_month unsigned, day_of_week unsigned, hour unsigned, minute unsigned, second unsigned, milliseconds long-unsigned }	日期和时间的十六进制格式。 year、milliseconds=FFFFH 表示无效。 month、day_of_month、day_of_week、hour、minute、second=FFH 表示无效。 day_of_week: 0 表示周日, 1...6 分别表示周一到周六。

7.3.27 date_time_s 数据类型

日期时间date_time_s的数据类型定义见表30。

表 30 date_time_s 数据类型定义

数据类型定义	说明
date_time_s ::= SEQUENCE { year long-unsigned, }	日期和时间的十六进制格式。 year=FFFFH 表示无效。 month、day、hour、minute、second=FFH 表示无效。

数据类型定义	说明
<pre> month unsigned, day unsigned, hour unsigned, minute unsigned, second unsigned } </pre>	

7.3.28 date 数据类型

日期date的数据类型定义见表31。

表 31 date 数据类型定义

数据类型定义	说明
<pre> date ::= SEQUENCE { year long-unsigned, month unsigned, day_of_month unsigned, day_of_week unsigned } </pre>	<p>日期的十六进制格式。 year=FFFFH 表示无效。 month、day_of_month、day_of_week=FFH 表示无效。</p>

7.3.29 time 数据类型

时间time的数据类型定义见表32。

表 32 time 数据类型定义

数据类型定义	说明
<pre> time ::= SEQUENCE { hour unsigned, minute unsigned, second unsigned } </pre>	<p>时间的十六进制格式。 hour、minute、second=FFH 表示无效。</p>

7.4 应用层数据单元

7.4.1 应用层数据单元定义

7.4.1.1 预连接数据单元

预连接数据单元（LINK-APDU）的数据类型定义见表33。

表 33 LINK-APDU 定义

数据类型定义	说明
<pre>LINK-APDU ::= CHOICE { 预连接请求 [1] LINK-Request, 预连接响应 [129] LINK-Response }</pre>	

7.4.1.2 客户机应用层数据单元

客户机应用层数据单元（Client-APDU）的数据类型定义见表34。

表 34 Client-APDU 定义

数据类型定义	说明
<pre>Client-APDU ::= SEQUENCE { 应用层服务 CHOICE { 建立应用连接请求 [2] CONNECT-Request, 断开应用连接请求 [3] RELEASE-Request, 读取请求 [5] GET-Request, 设置请求 [6] SET-Request, 操作请求 [7] ACTION-Request, 上报应答 [8] REPORT-Response, 代理请求 [9] PROXY-Request, 紧凑读取请求 [69] COMPACT-GET-Request, 紧凑设置请求 [70] COMPACT-SET-Request, 紧凑代理请求 [73] COMPACT-PROXY-Request, 异常响应 [110] ERROR-Response }, 时间标签域 TimeTag OPTIONAL }</pre>	

7.4.1.3 服务器应用层数据单元

服务器应用层数据单元（Server-APDU）的数据类型定义见表35。

表 35 Server-APDU 定义

数据类型定义	说明
<pre>Server-APDU ::= SEQUENCE { 应用层服务 CHOICE {</pre>	

数据类型定义		说明
建立应用连接响应	[130] CONNECT-Response,	
断开应用连接响应	[131] RELEASE-Response,	
断开应用连接通知	[132] RELEASE-Notification,	
读取响应	[133] GET-Response,	
设置响应	[134] SET-Response,	
操作响应	[135] ACTION-Response,	
上报通知	[136] REPORT-Notification,	
代理响应	[137] PROXY-Response,	
紧凑读取响应	[197] COMPACT-GET-Response,	
紧凑设置响应	[198] COMPACT-SET-Response,	
紧凑代理响应	[201] COMPACT-PROXY-Response,	
异常响应	[238] ERROR-Response	
},		
跟随上报信息域	FollowReport OPTIONAL,	
时间标签域	TimeTag OPTIONAL	
}		

7.4.1.4 安全传输数据单元

安全传输数据单元（SECURITY-APDU）的数据类型定义见表36。

表 36 SECURITY-APDU 定义

数据类型定义		说明
SECURITY-APDU ::= CHOICE		
{		
安全请求	[16] SECURITY-Request,	
安全响应	[144] SECURITY-Response	
}		

7.4.2 预连接

7.4.2.1 LINK-Request 数据类型

预连接请求（LINK-Request）的数据类型定义见表37。

表 37 LINK-Request 数据类型定义

数据类型定义		说明
LINK-Request ::= SEQUENCE		
{		
服务序号-优先级-ACD	PIID-ACD	PIID-ACD——见 0。
请求类型	ENUMERATED	心跳周期——单位：秒。
{		
登录	(0),	date_time——见 0。
}		

数据类型定义	说明
<pre>心跳 (1), 退出登录 (2) }, 心跳周期 long-unsigned, 请求时间 date_time, }</pre>	

7.4.2.2 LINK-Response 数据类型

预连接响应（LINK-Response）的数据类型定义见表38。

表 38 LINK-Response 数据类型定义

数据类型定义	说明
<pre>LINK-Response ::= SEQUENCE { 服务序号-优先级 PIID, 结果 Result, 请求时间 date_time, 收到时间 date_time, 响应时间 date_time, }</pre>	<p>PIID——见 7.3.3。</p> <p>date_time——见 0。</p> <p>时钟可信标志——用于表示响应方的时钟是否可信，bit7=0：不可信，bit7=1：可信。</p> <p>bit0…bit2——二进制编码表示结果，0：成功，1：地址重复，2：非法设备，3：容量不足，其它值：保留。</p>
<pre>Result ::= bit-string (SIZE (8)) { bit7 (时钟可信标志) (0), bit6 (保留) (1), bit5 (保留) (2), bit4 (保留) (3), bit3 (保留) (4), bit2 (结果) (5), bit1 (结果) (6), bit0 (结果) (7) }</pre>	

7.4.3 建立应用连接

7.4.3.1 密钥协商

在建立应用连接时进行密钥协商，产生会话密钥，用于计算数据验证码和链路用户数据的加密。

7.4.3.2 CONNECT-Request 数据类型

请求建立应用连接（CONNECT-Request）的数据类型定义见表39。

表 39 CONNECT-Request 数据类型定义

数据类型定义	说明
Connect-Request ::= SEQUENCE { 服务序号-优先级 PIID, 期望的协议版本号 long-unsigned, 期望的协议一致性块 ProtocolConformance, 期望的功能一致性块 FunctionConformance, 客户机发送帧最大尺寸 long-unsigned, 客户机接收帧最大尺寸 long-unsigned, 客户机接收帧最大窗口尺寸 unsigned, 客户机最大可处理帧尺寸 long-unsigned, 期望的应用连接超时时间 double-long-unsigned, 认证请求对象 ConnectMechanismInfo }	PIID——见 7.3.3。 ProtocolConformance——定义见附录 C。 FunctionConformance——定义见附录 C。 客户机发送帧最大尺寸——单位：字节。 客户机接收帧最大尺寸——单位：字节。 客户机接收帧最大窗口尺寸——单位：个。 期望的应用连接超时时间——单位：秒。

7.4.3.3 CONNECT-Response 数据类型

建立应用连接响应（CONNECT-Response）的数据类型定义见表40。

表 40 CONNECT-Response 数据类型定义

数据类型定义	说明
CONNECT-Response ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 服务器厂商版本信息 FactoryVersion, 商定的协议版本号 long-unsigned, 商定的协议一致性块 ProtocolConformance, 商定的功能一致性块 FunctionConformance, 服务器发送帧最大尺寸 long-unsigned, 服务器接收帧最大尺寸 long-unsigned, 服务器接收帧最大窗口尺寸 unsigned, 服务器最大可处理帧尺寸 long-unsigned, 商定的应用连接超时时间 double-long-unsigned, 连接响应对象 ConnectResponseInfo }	PIID-ACD——见 0。 ProtocolConformance——定义见附录 C。 FunctionConformance——定义见附录 C。 服务器发送帧最大尺寸——单位：字节。 服务器接收帧最大尺寸——单位：字节。 服务器接收帧最大窗口尺寸——单位：个。
FactoryVersion ::= SEQUENCE { 厂商代码 visible-string(SIZE(4)), 软件版本号 visible-string(SIZE(4)), 软件版本日期 visible-string(SIZE(6)), 硬件版本号 visible-string(SIZE(4)), 硬件版本日期 visible-string(SIZE(6)), 厂商扩展信息 visible-string(SIZE(8)) }	

数据类型定义	说明
}	

7.4.4 断开应用连接

7.4.4.1 RELEASE-Request 数据类型

请求断开应用连接（RELEASE-Request）的数据类型定义见表41。

表 41 RELEASE-Request 数据类型定义

数据类型定义	说明
RELEASE-Request ::= SEQUENCE { 服务序号-优先级 PIID }	PIID——见 7.3.3。

7.4.4.2 RELEASE-Response 数据类型

断开应用连接响应（RELEASE-Response）的数据类型定义见表42。

表 42 RELEASE-Response 数据类型定义

数据类型定义	说明
RELEASE-Response ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 结果 ENUMERATED {成功 (0)} }	PIID-ACD——见 0。

7.4.4.3 RELEASE-Notification 数据类型

断开应用连接通知（RELEASE-Notification）的数据类型定义见表43。

表 43 RELEASE-Notification 数据类型定义

数据类型定义	说明
RELEASE-Notification ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 应用连接建立时间 date_time_s, 服务器当前时间 date_time_s }	PIID-ACD——见 0。 date_time_s——见 0。

7.4.5 读取

7.4.5.1 GET-Request 数据类型

7.4.5.1.1 概述

请求读取（GET-Request）的数据类型定义见表44。

表 44 GET-Request 数据类型定义

数据类型定义	说明
GET-Request ::= CHOICE { 请求读取一个对象属性 [1] GetRequestNormal, 请求读取若干个对象属性 [2] GetRequestNormalList, 请求读取一个记录型对象属性 [3] GetRequestRecord, 请求读取若干个记录型对象属性 [4] GetRequestRecordList, 请求读取分帧传输的下一帧 [5] GetRequestNext, 请求读取一个对象属性的 MD5 值 [6] GetRequestMD5, 读取一个精简的记录型对象属性请求 [23] GetRequestSimplifyRecord }	

7.4.5.1.2 GetRequestNormal 数据类型

请求读取一个对象属性（GetRequestNormal）的数据类型定义见表45。

表 45 GetRequestNormal 数据类型定义

数据类型定义	说明
GetRequestNormal ::= SEQUENCE { 服务序号-优先级 PIID, 一个对象属性描述符 OAD }	PIID——见 7.3.3。 OAD——见 7.3.6。

7.4.5.1.3 GetRequestNormalList 数据类型

请求读取若干个对象属性（GetRequestNormalList）的数据类型定义见表46。

表 46 GetRequestNormalList 数据类型定义

数据类型定义	说明
GetRequestNormalList ::= SEQUENCE { 服务序号-优先级 PIID, 若干个对象属性描述符 SEQUENCE OF OAD }	PIID——见 7.3.3。 OAD——见 7.3.6。

7.4.5.1.4 GetRequestRecord 数据类型

请求读取一个记录型对象属性（GetRequestRecord）的数据类型定义见表47。

表 47 GetRequestRecord 数据类型定义

数据类型定义	说明
GetRequestRecord ::= SEQUENCE { 服务序号-优先级 PIID, 读取一个记录型对象属性 GetRecord }	PIID——见 7.3.3。 OAD——见 7.3.6。 RSD——见 0。 RCSD——见 7.3.21。
GetRecord ::= SEQUENCE { 对象属性描述符 OAD, 记录行选择描述符 RSD, 记录列选择描述符 RCSD }	

7.4.5.1.5 GetRequestRecordList 数据类型

请求读取若干个记录型对象属性（GetRequestRecordList）的数据类型定义见表48。

表 48 GetRequestRecordList 数据类型定义

数据类型定义	说明
GetRequestRecordList ::= SEQUENCE { 服务序号-优先级 PIID, 读取若干个记录型对象属性 SEQUENCE OF GetRecord }	PIID——见 7.3.3。 GetRecord——见表 47。

7.4.5.1.6 GetRequestNext 数据类型

请求读取分帧传输的下一帧（GetRequestNext）的数据类型定义见表49。

表 49 GetRequestNext 数据类型定义

数据类型定义	说明
GetRequestNext ::= SEQUENCE { 服务序号-优先级 PIID, 已接收的最后分帧序号 long-unsigned }	PIID——见 7.3.3。 服务器应答分帧序号为“已接收的最后分帧序号+1”。

7.4.5.1.7 GetRequestMD5 数据类型

请求读取一个对象属性MD5值（GetRequestMD5）的数据类型定义见表50。

表 50 GetRequestMD5 数据类型定义

数据类型定义	说明
GetRequestMD5 ::= SEQUENCE { 服务序号-优先级 PIID, 一个对象属性描述符 OAD }	PIID——见 7.3.3 。

7.4.5.1.8 GetRequestSimplifyRecord 数据类型

读取一个精简的记录型对象属性请求（GetRequestSimplifyRecord）的数据类型定义见表51。

表 51 GetRequestSimplifyRecord 数据类型定义

数据类型定义	说明
GetRequestSimplifyRecord ::= SEQUENCE { 服务序号-优先级 PIID, 读取一个记录型对象属性 GetConsistentRecord }	PIID——见 7.3.3 。
GetConsistentRecord ::= SEQUENCE { 对象属性描述符 OAD, 记录选择描述符 RSD, 记录列选择描述符 ROAD }	

7.4.5.2 GET-Response 数据类型

7.4.5.2.1 概述

响应读取（GET-Response）的数据类型定义见表52。

表 52 GET-Response 数据类型定义

数据类型定义	说明
GET-Response ::= CHOICE { 读取一个对象属性的响应 [1] GetResponseNormal, 读取若干个对象属性的响应 [2] GetResponseNormalList, 读取一个记录型对象属性的响应 [3] GetResponseRecord, 读取若干个记录型对象属性的响应 [4] GetResponseRecordList, 读取分帧传输的下一帧的响应 [5] GetResponseNext, 读取一个对象属性的 MD5 值的响应 [6] GetResponseMD5, 读取一个精简记录型对象属性请求 [23] GetResponseSimplifyRecord }	

7.4.5.2.2 GetResponseNormal 数据类型

响应读取一个对象属性（GetResponseNormal）的数据类型定义见表53。

表 53 GetResponseNormal 数据类型定义

数据类型定义	说明
GetResponseNormal ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 一个对象属性及结果 A-ResultNormal }	PIID-ACD——见 0。
A-ResultNormal ::= SEQUENCE { 对象属性描述符 OAD, 读取结果 Get-Result }	OAD——见 7.3.6。
Get-Result ::= CHOICE { 错误信息 [0] DAR, 数据 [1] Data }	DAR——见 7.3.22。 Data——见 7.3.1。

7.4.5.2.3 GetResponseNormalList 数据类型

响应读取若干个对象属性（GetResponseNormalList）的数据类型定义见表54。

表 54 GetResponseNormalList 数据类型定义

数据类型定义	说明
GetResponseNormalList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 若干个对象属性及结果 SEQUENCE OF A-ResultNormal }	PIID-ACD——见 0。 A-ResultNormal——见表 53。

7.4.5.2.4 GetResponseRecord 数据类型

响应读取一个记录型对象属性（GetResponseRecord）的数据类型定义见表55。

表 55 GetResponseRecord 数据类型定义

数据类型定义	说明
GetResponseRecord ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, }	PIID-ACD——见 0。

数据类型定义	说明
一个记录型对象属性及结果 A-ResultRecord }	
A-ResultRecord ::= SEQUENCE { 记录型对象属性描述符 OAD, 记录的 N 列属性描述符 RCSD, 响应数据 CHOICE { 错误信息 [0] DAR, M 条记录 [1] SEQUENCE OF A-RecordRow } }	OAD——见 7.3.6。 记录的 N 列属性描述符——记录表的表头信息。
A-RecordRow ::= SEQUENCE { 第 1 列数据 Data, 第 2 列数据 Data, ... 第 N 列数据 Data }	第 1 列…第 N 列的数据，其排列顺序与“记录的 N 列属性描述符”的排列顺序一致。 Data——见 7.3.1。

7.4.5.2.5 GetResponseRecordList 数据类型

响应读取若干个记录型对象属性（GetResponseRecordList）的数据类型定义见表 56。

表 56 GetResponseRecordList 数据类型定义

数据类型定义	说明
GetResponseRecordList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 若干个记录型对象属性及结果 SEQUENCE OF A-ResultRecord }	PIID-ACD——见 0。 OAD——见 7.3.6。 A-ResultRecord——见表 55。

7.4.5.2.6 GetResponseNext 数据类型

响应读取分帧传输的下一帧（GetResponseNext）的数据类型定义见表 57。

表 57 GetResponseNext 数据类型定义

数据类型定义	说明
GetResponseNext ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 末帧标志 bool,	PIID-ACD——见 0。 DAR——见 7.3.22。 A-ResultNormal——见表 53。 A-ResultRecord——见表 55。

数据类型定义		说明
分帧序号	long-unsigned,	
分帧响应	CHOICE	
{		
错误信息	[0] DAR,	
对象属性	[1] SEQUENCE OF A-ResultNormal,	
记录型对象属性	[2] SEQUENCE OF A-ResultRecord	
}		
}		

7.4.5.2.7 GetResponseMD5 数据类型

响应读取对象属性MD5值（GetResponseMD5）的数据类型定义见表58。

表 58 GetResponseMD5 数据类型定义

数据类型定义		说明
GetResponseMD5 ::= SEQUENCE		PIID-ACD——见 0。 DAR——见 7.3.22。 MD5 值——使用编码后的对象属性值进行计算。
{		
服务序号-优先级-ACD	PIID-ACD,	
对象属性描述符	OAD,	
结果	CHOICE	
{		
错误信息	[0] DAR,	
MD5 值	[1] octet-string	
}		
}		

7.4.5.2.8 GetResponseSimplifyRecord 数据类型

响应读取一个精简的记录型对象属性（GetResponseSimplifyRecord）的数据类型定义见表59。

表 59 GetResponseSimplifyRecord 数据类型定义

数据类型定义		说明
GetResponseSimplifyRecord ::= SEQUENCE		PIID-ACD——见 0。 DAR——见 7.3.22。
{		
服务序号-优先级-ACD	PIID-ACD,	
一个精简的记录型对象属性及其结果	A-ResultSimplifyRecord	
}		
A-ResultSimplifyRecord ::= SEQUENCE		
{		
记录型对象属性描述符	OAD,	
记录选择描述符	RSD,	
一行记录 N 列属性描述符	ROAD,	
响应数据	CHOICE	

数据类型定义	说明
<pre> { 错误信息 [0] DAR, 记录 [1] A-SimplifyRecord } </pre>	
<pre> A-SimplifyRecord ::= SEQUENCE { 记录列数据类型 A-SimplifyRecordRowDataType M 条记录数据值 SEQUENCE OF A-SimplifyRecordRow } A-SimplifyRecordRowDataType ::= SEQUENCE { 第 1 列数据类型 CHOICE, 第 2 列数据类型 CHOICE, ... 第 N 列数据类型 CHOICE } A-SimplifyRecordRow ::= SEQUENCE { 第 1 列数据值, 第 2 列数据值, ... 第 N 列数据值 } </pre>	

7.4.6 设置

7.4.6.1 SET-Request 数据类型

7.4.6.1.1 概述

设置请求（SET-Request）的数据类型定义见表60。

表 60 SET-Request 数据类型定义

数据类型定义	说明
<pre> SET-Request ::= CHOICE { 请求设置一个对象属性 [1] SetRequestNormal, 请求设置若干个对象属性 [2] SetRequestNormalList, 请求设置后读取若干个对象属性 [3] SetThenGetRequestNormalList } </pre>	

7.4.6.1.2 SetRequestNormal 数据类型

请求设置一个对象属性（SetRequestNormal）的数据类型定义见表61。

表 61 SetRequestNormal 数据类型定义

数据类型定义	说明
SetRequestNormal ::= SEQUENCE { 服务序号-优先级 PIID, 一个对象属性描述符 OAD, 数据 Data }	PIID——见 7.3.3。 OAD——见 7.3.6。 Data——见 7.3.1。

7.4.6.1.3 SetRequestNormalList 数据类型

请求设置若干个对象属性（SetRequestNormalList）的数据类型定义见表62。

表 62 SetRequestNormalList 数据类型定义

数据类型定义	说明
SetRequestNormalList ::= SEQUENCE { 服务序号-优先级 PIID, 若干个对象属性 SEQUENCE OF { 一个对象属性描述符 OAD, 数据 Data } }	PIID——见 7.3.3。 OAD——见 7.3.6。 Data——见 7.3.1。

7.4.6.1.4 SetThenGetRequestNormalList 数据类型

请求设置后读取若干个对象属性（SetThenGetRequestNormalList）的数据类型定义见表63。

表 63 SetThenGetRequestNormalList 数据类型定义

数据类型定义	说明
SetThenGetRequestNormalList ::= SEQUENCE { 服务序号-优先级 PIID, 若干个设置后读取对象属性 SEQUENCE OF { 一个设置的对象属性 OAD, 数据 Data, 延时读取时间——单位：秒，0 表示 由服务器确定延时时间。 } }	PIID——见 7.3.3。 OAD——见 7.3.6。 Data——见 7.3.1。 延时读取时间——单位：秒，0 表示由服务器确定延时时间。

数据类型定义	说明
一个读取的对象属性 OAD, 延时读取时间 unsigned } }	

7.4.6.2 SET-Response 数据类型

7.4.6.2.1 概述

响应设置（SET-Response）的数据类型定义见表64。

表 64 SET-Response 数据类型定义

数据类型定义	说明
SET-Response ::= CHOICE { 设置一个对象属性的响应 [1] SetResponseNormal, 设置若干个对象属性的响应 [2] SetResponseNormalList, 设置的确认信息以及读取的响应 [3] SetThenGetResponseNormalList }	

7.4.6.2.2 SetResponseNormal 数据类型

响应设置一个对象属性（SetResponseNormal）的数据类型定义见表65。

表 65 SetResponseNormal 数据类型定义

数据类型定义	说明
SetResponseNormal ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 一个对象属性描述符 OAD, 设置执行结果 DAR }	PIID-ACD——见 0。 OAD——见 7.3.6。 DAR——见 7.3.22。

7.4.6.2.3 SetResponseNormalList 数据类型

响应设置若干个对象属性（SetResponseNormalList）的数据类型定义见表66。

表 66 SetResponseNormalList 数据类型定义

数据类型定义	说明
SetResponseNormalList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 若干个对象属性设置结果 SEQUENCE OF	PIID-ACD——见 0。 OAD——见 7.3.6。 DAR——见 7.3.22。

数据类型定义	说明
<pre> { 一个对象属性描述符 OAD, 设置执行结果 DAR } </pre>	

7.4.6.2.4 SetThenGetResponseNormalList 数据类型

响应设置若干个对象属性以及读取若干个对象属性（SetThenGetResponseNormalList）的数据类型定义见表67。

表 67 SetThenGetResponseNormalList 数据类型定义

数据类型定义	说明
<pre> SetThenGetResponseNormalList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 若干个对象属性设置后读取结果 SEQUENCE OF { 一个设置的对象属性描述符 OAD, 设置执行结果 DAR, 一个对象属性及结果 A-ResultNormal } } </pre>	<p>PIID-ACD——见 0。</p> <p>OAD——见 7.3.6。</p> <p>DAR——见 7.3.22。</p> <p>A-ResultNormal——见表 53。</p>

7.4.7 操作

7.4.7.1 ACTION-Request 数据类型

7.4.7.1.1 ACTION-Request 数据类型定义

请求操作（ACTION-Request）的数据类型定义见表68。

表 68 ACTION-Request 数据类型定义

数据类型定义	说明
<pre> ACTION-Request ::= CHOICE { 请求操作一个对象方法 [1] ActionRequestNormal, 请求操作若干个对象方法 [2] ActionRequestNormalList, 请求操作若干个对象方法后读取若干个对象属性 [3] ActionThenGetRequestNormalList } </pre>	

7.4.7.1.2 ActionRequestNormal 数据类型

请求操作一个对象方法（ActionRequestNormal）的数据类型定义见表69。

表 69 ActionRequestNormal 数据类型定义

数据类型定义	说明
<pre> ActionRequestNormal ::= SEQUENCE { 服务序号-优先级 PIID, 一个对象方法描述符 OMD, 方法参数 Data } </pre>	PIID——见 7.3.3 。 OMD——见 7.3.8 。 Data——见 7.3.1 。

7.4.7.1.3 ActionRequestNormalList 数据类型

请求操作若干个对象方法（ActionRequestNormalList）的数据类型定义见表70。

表 70 ActionRequestNormalList 数据类型定义

数据类型定义	说明
<pre> ActionRequestNormalList ::= SEQUENCE { 服务序号-优先级 PIID, 若干个对象方法 SEQUENCE OF { 一个对象方法描述符 OMD, 方法参数 Data } } </pre>	PIID——见 7.3.3 。 OMD——见 7.3.8 。 Data——见 7.3.1 。

7.4.7.1.4 ActionThenGetRequestNormalList 数据类型

请求操作若干个对象方法后读取若干个对象属性（ActionThenGetRequestNormalList）的数据类型定义见表71。

表 71 ActionThenGetRequestNormalList 数据类型定义

数据类型定义	说明
<pre> ActionThenGetRequestNormalList ::= SEQUENCE { 服务序号-优先级 PIID, 操作若干个对象方法后读取对象属性 SEQUENCE OF { 一个设置的对象方法描述符 OMD, 方法参数 Data, 一个读取的对象属性描述符 OAD, 读取延时 unsigned } } </pre>	PIID——见 7.3.3 。 OAD——见 7.3.6 。 OMD——见 7.3.8 。 Data——见 7.3.1 。 读取延时——单位：秒，0 表示由服务器确定延时时间。

7.4.7.2 ACTION-Response 数据类型

7.4.7.2.1 Action-Response 数据类型定义

响应操作（ACTION-Response）的数据类型定义见表72。

表 72 ACTION-Response 数据类型定义

数据类型定义	说明
ACTION-Response ::= CHOICE { 操作一个对象方法的响应 [1] ActionResponseNormal, 操作若干个对象方法的响应 [2] ActionResponseNormalList, 操作若干个对象方法后读取若干个属性的响应 [3] ActionThenGetResponseNormalList }	

7.4.7.2.2 ActionResponseNormal 数据类型

响应操作一个对象方法（ActionResponseNormal）的数据类型定义见表73。

表 73 ActionResponseNormal 数据类型定义

数据类型定义	说明
ActionResponseNormal ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 一个对象方法描述符 OMD, 操作执行结果 DAR, 操作返回数据 Data OPTIONAL }	PIID-ACD——见 0。 OMD——见 7.3.8。 DAR——见 7.3.22。 Data——见 7.3.1。

7.4.7.2.3 ActionResponseNormalList 数据类型

响应操作若干个对象方法（ActionResponseNormalList）的数据类型定义见表74。

表 74 ActionResponseNormalList 数据类型定义

数据类型定义	说明
ActionResponseNormalList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 若干个对象方法操作结果 SEQUENCE OF { 一个对象方法描述符 OMD, 操作执行结果 DAR, 操作返回数据 Data OPTIONAL } }	PIID-ACD——见 0。 OMD——见 7.3.8。 DAR——见 7.3.22。 Data——见 7.3.1。

7.4.7.2.4 ActionThenGetResponseNormalList 数据类型

响应操作若干个对象方法后读取若干个属性（ActionThenGetResponseNormalList）的数据类型定义见表75。

表 75 ActionThenGetResponseNormalList 数据类型定义

数据类型定义	说明
<pre> ActionThenGetResponseNormalList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 操作若干个对象方法后读取属性的结果 SEQUENCE OF { 一个设置的对象方法描述符 OMD, 操作执行结果 DAR, 操作返回数据 Data OPTIONAL, 一个对象属性及结果 A-ResultNormal } } </pre>	PIID-ACD——见 0。 OMD——见 7.3.8。 DAR——见 7.3.22。 OAD——见 7.3.6。 Data——见 7.3.1。 A-ResultNormal——见表 53。

7.4.8 上报

7.4.8.1 REPORT-Notification 数据类型

7.4.8.1.1 REPORT-Notification 数据类型定义

通知上报（REPORT-Notification）的数据类型定义见表76。

表 76 REPORT-Notification 数据类型定义

数据类型定义	说明
<pre> REPORT-Notification ::= CHOICE { 通知上报若干个对象属性 [1] ReportNotificationList, 通知上报若干个记录型对象属性 [2] ReportNotificationRecordList, 通知上报透明数据 [3] ReportNotificationTransData, 请求客户机服务 [4] ReportRequestClientService, 通知上报精简的记录型对象属性 [6] ReportNotificationSimplifyRecord } </pre>	

7.4.8.1.2 ReportNotificationList 数据类型

通知上报若干个对象属性（ReportNotificationList）的数据类型定义见表77。

表 77 ReportNotificationList 数据类型定义

数据类型定义	说明
ReportNotificationList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 若干个对象属性及数据 SEQUENCE OF A-ResultNormal }	PIID-ACD——见 0。 A-ResultNormal——见表 53 。

7.4.8.1.3 ReportNotificationRecordList 数据类型

通知上报若干个记录型对象属性（ReportNotificationRecordList）的数据类型定义见表78 。

表 78 ReportNotificationRecordList 数据类型定义

数据类型定义	说明
ReportNotificationRecordList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 若干个记录型对象属性及数据 SEQUENCE OF A-ResultRecord }	PIID-ACD——见 0。 A-ResultRecord——见表 55 。

7.4.8.1.4 ReportNotificationTransData 数据类型

通知上报透明数据（ReportNotificationTransData）的数据类型定义见表79 。

表 79 ReportNotificationTransData 数据类型定义

数据类型定义	说明
ReportNotificationTransData ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 数据来源端口号 OAD, 透明数据 SEQUENCE OF octet-string }	仅用于终端将来自其通信端口的上报数据上报到客户机。 PIID-ACD——见 0。

7.4.8.1.5 ReportRequestClientService 数据类型

通知请求客户机服务（ReportRequestClientService）的数据类型定义见表80 。

表 80 ReportRequestClientService 数据类型定义

数据类型定义	说明
ReportRequestClientService ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 服务数据 octet-string }	用于终端请求客户机服务数据，数据内容为 Client-APDU。 PIID-ACD——见 0。

7.4.8.1.6 ReportNotificationSimplifyRecord 数据类型

通知上报精简的记录型对象属性（ReportNotificationSimplifyRecord）的数据类型定义见表81。

表81 ReportNotificationSimplifyRecord 数据类型定义

数据类型定义	说明
ReportNotificationSimplifyRecord ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 精简的记录型对象属性及数据 A-ResultSimplifyRecord }	

7.4.8.2 REPORT-Response 数据类型

7.4.8.2.1 REPORT-Response 数据类型定义

响应上报（REPORT-Response）的数据类型定义见表82。

表82 REPORT-Response 数据类型定义

数据类型定义	说明
REPORT-Response ::= CHOICE { 上报若干个对象属性的响应 [1] ReportResponseList, 上报若干个记录型对象属性的响应 [2] ReportResponseRecordList, 上报透明数据的响应 [3] ReportResponseTransData, 请求客户机服务的响应 [4] ReportResponseClientService, 上报精简的记录型对象属性的响应 [6] ReportResponseSimplifyRecord }	

7.4.8.2.2 ReportResponseList 数据类型

响应上报若干个对象属性（ReportResponseList）的数据类型定义见表83。

表83 ReportResponseList 数据类型定义

数据类型定义	说明
ReportResponseList ::= SEQUENCE { 服务序号-优先级 PIID, 对应上报的若干个对象属性描述符 SEQUENCE OF OAD }	对应上报的若干个对象属性描述符:用于向服务器表明接收确认。 PIID-ACD——见0。 OAD——见7.3.6。

7.4.8.2.3 ReportResponseRecordList 数据类型

响应上报若干个记录型对象属性（ReportResponseRecordList）的数据类型定义见表84。

表 84 ReportResponseRecordList 数据类型定义

数据类型定义	说明
ReportResponseRecordList ::= SEQUENCE { 服务序号-优先级 PIID, 对应上报的若干个对象属性描述符 SEQUENCE OF OAD }	对应上报的若干个对象属性描述符:用于向服务器表明接收确认。 PIID——见 7.3.3。 OAD——见 7.3.6。

7.4.8.2.4 ReportResponseTransData 数据类型

响应上报透明数据（ReportResponseTransData）的数据类型定义见表85。

表 85 ReportResponseTransData 数据类型定义

数据类型定义	说明
ReportResponseTransData ::= SEQUENCE { 服务序号-优先级 PIID }	PIID——见 7.3.3。

7.4.8.2.5 ReportResponseClientService 数据类型

响应请求客户机服务（ReportResponseClientService）的数据类型定义见表86。

表 86 ReportResponseClientService 数据类型定义

数据类型定义	说明
ReportResponseClientService ::= SEQUENCE { 服务序号-优先级 PIID, 服务数据 octet-string }	用于终端请求客户机服务数据的响应，数据内容为 Server-APDU。 PIID——见 7.3.3。

7.4.8.2.6 ReportResponseSimplifyRecord 数据类型

响应上报精简的记录型对象属性服务（ReportResponseSimplifyRecord）的数据类型定义见表87。

表 87 ReportResponseSimplifyRecord 数据类型定义

数据类型定义	说明
ReportResponseSimplifyRecord ::= SEQUENCE {	PIID——见 7.3.3。

服务序号-优先级	PIID	
}		

7.4.9 代理

7.4.9.1 PROXY-Request 数据类型

7.4.9.1.1 概述

请求代理（PROXY-Request）的数据类型定义见表88。

表 88 PROXY-Request 数据类型定义

数据类型定义	说明
PROXY-Request ::= CHOICE { 请求代理读取若干个服务器的若干个对象属性 [1] ProxyGetRequestList, 请求代理读取一个服务器的一个记录型对象属性 [2] ProxyGetRequestRecord, 请求代理设置若干个服务器的若干个对象属性 [3] ProxySetRequestList, 请求代理设置后读取若干个服务器的若干个对象属性 [4] ProxySetThenGetRequestList, 请求代理操作若干个服务器的若干个对象方法 [5] ProxyActionRequestList, 请求代理操作后读取若干个服务器的若干个对象方法和属性 [6] ProxyActionThenGetRequestList, 请求代理透明转发命令 [7] ProxyTransCommandRequest 请求代理终端内部转发命令 [8] ProxyInnerTransCommandRequest }	

7.4.9.1.2 ProxyGetRequestList 数据类型

请求代理读取若干个服务器的若干个对象属性（ProxyGetRequestList）的数据类型定义见表89。

表 89 ProxyGetRequestList 数据类型定义

数据类型定义	说明
ProxyGetRequestList ::= SEQUENCE { 服务序号-优先级 PIID, 代理整个请求的超时时间 long-unsigned, 代理若干个服务器的对象属性读取 SEQUENCE OF { 一个目标服务器地址 TSA, 代理一个目标服务器的超时时间 long-unsigned, 若干个对象属性描述符 SEQUENCE OF OAD } }	PIID——见 7.3.3。 TSA——见 7.3.10。 OAD——见 7.3.6。 代理整个请求的超时时间——单位：秒，非 0 值。 代理一个目标服务器的超时时间——单位：秒，0 表示由代理服务器确定超时时间。

7.4.9.1.3 ProxyGetRequestRecord 数据类型

请求代理读取一个服务器的一个记录型对象属性（ProxyGetRequestRecord）的数据类型定义见表90。

表 90 ProxyGetRequestRecord 数据类型定义

数据类型定义	说明
<pre>ProxyGetRequestRecord ::= SEQUENCE { 服务序号-优先级 PIID, 代理请求的超时时间 long-unsigned, 目标服务器地址 TSA, 对象属性描述符 OAD, 记录行选择描述符 RSD, 记录列选择描述符 RCSD }</pre>	PIID——见 7.3.3。 TSA——见 7.3.10。 OAD——见 7.3.6。 RSD——见 0。 RCSD——见 7.3.21。 代理请求的超时时间——单位：秒，非 0 值。

7.4.9.1.4 ProxySetRequestList 数据类型

请求代理设置若干个服务器的若干个对象属性（ProxySetRequestList）的数据类型定义见表91。

表 91 ProxySetRequestList 数据类型定义

数据类型定义	说明
<pre>ProxySetRequestList ::= SEQUENCE { 服务序号-优先级 PIID, 代理整个请求的超时时间 long-unsigned, 代理若干个服务器的对象属性设置 SEQUENCE OF { 一个目标服务器地址 TSA, 代理一个目标服务器的超时时间 long-unsigned, 若干个对象属性描述符及数据 SEQUENCE OF { 对象属性描述符 OAD, 设置数值 Data } } }</pre>	PIID——见 7.3.3。 TSA——见 7.3.10。 OAD——见 7.3.6。 Data——见 7.3.1。 代理整个请求的超时时间——单位：秒，非 0 值。 代理一个目标服务器的超时时间——单位：秒，0 表示由代理服务器确定超时时间。

7.4.9.1.5 ProxySetThenGetRequestList 数据类型

请求代理设置后读取若干个服务器的若干个对象属性（ProxySetThenGetRequestList）的数据类型定义见表92。

表 92 ProxySetThenGetRequestList 数据类型定义

数据类型定义	说明
--------	----

数据类型定义	说明
<pre>ProxySetThenGetRequestList ::= SEQUENCE { 服务序号-优先级 PIID, 代理整个请求的超时时间 long-unsigned, 代理若干个服务器的对象属性设置后读取 SEQUENCE OF { 一个目标服务器地址 TSA, 代理一个目标服务器的超时时间 long-unsigned, 若干个对象属性的设置后读取 SEQUENCE OF { 设置的对象属性描述符 OAD, 设置数值 Data, 读取的对象属性描述符 OAD, 延时读取时间 unsigned } } }</pre>	<p>PIID——见 7.3.3 。</p> <p>TSA——见 7.3.10 。</p> <p>OAD——见 7.3.6 。</p> <p>Data——见 7.3.1 。</p> <p>代理整个请求的超时时间——单位：秒，非 0 值。</p> <p>代理一个目标服务器的超时时间——单位：秒，0 表示由代理服务器确定超时时间。</p> <p>延时读取时间——单位：秒，0 表示由目标服务器确定延时读取时间。</p>

7.4.9.1.6 ProxyActionRequestList 数据类型

请求代理操作若干个服务器的若干个对象方法（ProxyActionRequestList）数据类型定义见表 93 。

表 93 ProxyActionRequestList 数据类型定义

数据类型定义	说明
<pre>ProxyActionRequestList ::= SEQUENCE { 服务序号-优先级 PIID, 代理整个请求的超时时间 long-unsigned, 代理若干个服务器的对象方法操作 SEQUENCE OF { 一个目标服务器地址 TSA, 代理一个目标服务器的超时时间 long-unsigned, 若干个对象方法描述符及参数 SEQUENCE OF { 对象方法描述符 OMD, 方法参数 Data } } }</pre>	<p>PIID——见 7.3.3 。</p> <p>TSA——见 7.3.10 。</p> <p>OMD——见 7.3.8 。</p> <p>Data——见 7.3.1 。</p> <p>代理整个请求的超时时间——单位：秒，非 0 值。</p> <p>代理一个目标服务器的超时时间——单位：秒，0 表示由代理服务器确定超时时间。</p>

7.4.9.1.7 ProxyActionThenGetRequestList 数据类型

请求代理操作后读取若干个服务器的若干个对象方法和属性（ProxyActionThenGetRequestList）的数据类型定义见表94。

表 94 ProxyActionThenGetRequestList 数据类型定义

数据类型定义	说明
<pre>ProxyActionThenGetRequestList ::= SEQUENCE { 服务序号-优先级 PIID, 代理整个请求的超时时间 long-unsigned, 代理若干个服务器的操作后读取 SEQUENCE OF { 一个目标服务器地址 TSA, 代理一个目标服务器的超时时间 long-unsigned, 若干个对象方法及属性的操作后读取 SEQUENCE OF { 操作的对象方法描述符 OMD, 方法参数 Data, 读取的对象属性描述符 OAD, 延时读取时间 unsigned } } }</pre>	<p>PIID——见 7.3.3。</p> <p>TSA——见 7.3.10。</p> <p>OMD——见 7.3.8。</p> <p>Data——见 7.3.1。</p> <p>OAD——见 7.3.6。</p> <p>代理整个请求的超时时间——单位：秒，非 0 值。</p> <p>代理一个目标服务器的超时时间——单位：秒，0 表示由代理服务器确定超时时间。</p> <p>延时读取时间——单位：秒，0 表示由目标服务器确定延时读取时间。</p>

7.4.9.1.8 ProxyTransCommandRequest 数据类型

请求代理操作透明转发（ProxyTransCommandRequest）的数据类型定义见表95。

表 95 ProxyTransCommandRequest 数据类型定义

数据类型定义	说明
<pre>ProxyTransCommandRequest ::= SEQUENCE { 服务序号-优先级 PIID, 转发端口对象描述符 OAD, 端口通信控制块 COMDCB, 接收等待报文超时时间（秒） long-unsigned, 接收等待字节超时时间（毫秒） long-unsigned, 转发至目标端口的命令 octet-string }</pre>	<p>PIID——见 7.3.3。</p> <p>OAD——见 7.3.6。</p> <p>COMDCB——见 7.3.20。</p>

7.4.9.1.9 ProxyInnerTransCommandRequest 数据类型

请求代理终端内部透明转发（ProxyInnerTransCommandRequest）的数据类型定义见表95。

表 96 ProxyInnerTransCommandRequest 数据类型定义

数据类型定义	说明
<pre>ProxyInnerTransCommandRequest ::= SEQUENCE { 服务序号-优先级 PIID, 终端内部转发命令 SEQUENCE OF COMMAND }</pre>	PIID——见 7.3.3。 OAD——见 7.3.6。 COMDCB——见 7.3.20。
<pre>COMMAND ::= SEQUENCE { 转发至逻辑设备地址 unsigned, 接收等待报文超时时间（秒） long-unsigned, 接收等待字节超时时间（毫秒） long-unsigned, 转发至目标地址的命令 octet-string }</pre>	

7.4.9.2 PROXY-Response 数据类型

7.4.9.2.1 概述

响应代理（PROXY-Response）的数据类型定义见表96。

表 97 PROXY-Response 数据类型定义

数据类型定义	说明
<pre>PROXY-Response ::= CHOICE { 代理读取若干个服务器的若干个对象属性的响应 [1] ProxyGetResponseList, 代理读取一个服务器的一个记录型对象属性的响应 [2] ProxyGetResponseRecord, 代理设置若干个服务器的若干个对象属性的响应 [3] ProxySetResponseList, 代理设置后读取若干个服务器的若干个对象属性的响应 [4] ProxySetThenGetResponseList, 代理操作若干个服务器的若干个对象方法的响应 [5] ProxyActionResponseList, 代理操作后读取若干个服务器的若干个对象方法和属性的响应 [6] ProxyActionThenGetResponseList, 代理透明转发命令的响应 [7] ProxyTransCommandResponse 代理终端内部转发命令的响应 [8] ProxyInnerTransCommandResponse }</pre>	

7.4.9.2.2 ProxyGetResponseList 数据类型

响应代理读取若干个服务器的若干个对象属性（ProxyGetResponseList）的数据类型定义见表97。

表 98 ProxyGetResponseList 数据类型定义

数据类型定义	说明
ProxyGetResponseList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 一个目标服务器地址 TSA, 代理若干个服务器的读取结果 SEQUENCE OF { 若干个对象属性及结果 SEQUENCE OF A-ResultNormal } }	PIID-ACD——见 0。 TSA——见 7.3.10。 OAD——见 7.3.6。 A-ResultNormal——见表 53。

7.4.9.2.3 ProxyGetResponseRecord 数据类型

响应代理读取一个服务器的一个记录型对象属性 (ProxyGetResponseRecord) 的数据类型定义见表 98。

表 99 ProxyGetResponseRecord 数据类型定义

数据类型定义	说明
ProxyGetResponseRecord ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 目标服务器地址 TSA, 一个记录型对象属性及结果 A-ResultRecord }	PIID-ACD——见 0。 TSA——见 7.3.10。 A-ResultRecord——见表 55。

7.4.9.2.4 ProxySetResponseList 数据类型

响应代理设置若干个服务器的若干个对象属性 (ProxySetResponseList) 的数据类型定义见表 99。

表 100 ProxySetResponseList 数据类型定义

数据类型定义	说明
ProxySetResponseList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 代理若干个服务器的读取结果 SEQUENCE OF { 一个目标服务器地址 TSA, } }	PIID-ACD——见 0。 TSA——见 7.3.10。 OAD——见 7.3.6。 DAR——见 7.3.22。

数据类型定义	说明
若干个对象属性描述符及结果 SEQUENCE OF { 对象属性描述符 OAD, 设置结果 DAR } } }	

7.4.9.2.5 ProxySetThenGetResponseList 数据类型

响应代理设置后读取若干个服务器的若干个对象属性（ProxySetThenGetResponseList）的数据类型定义见表100。

表 101 ProxySetThenGetResponseList 数据类型定义

数据类型定义	说明
ProxySetThenGetResponseList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 代理若干个服务器的设置后读取结果 SEQUENCE OF { 一个目标服务器地址 TSA, 若干个对象属性设置后读取结果 SEQUENCE OF { 设置的对象属性描述符 OAD, 设置结果 DAR, 一个对象属性及结果 A-ResultNormal } } }	PIID-ACD——见 0。

7.4.9.2.6 ProxyActionResponseList 数据类型

响应代理操作若干个服务器的若干个对象方法（ProxyActionResponseList）的数据类型定义见表 101。

表 102 ProxyActionResponseList 数据类型定义

数据类型定义	说明
--------	----

数据类型定义	说明
<pre>ProxyActionResponseList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 代理若干个服务器的操作结果 SEQUENCE OF { 一个目标服务器地址 TSA, 若干个对象方法描述符及结果 SEQUENCE OF { 对象方法描述符 OMD, 操作结果 DAR, 操作返回数据 Data OPTIONAL } } }</pre>	<p>PIID-ACD——见 0。</p> <p>TSA——见 7.3.10 。</p> <p>OMD——见 7.3.8 。</p> <p>DAR——见 7.3.22 。</p> <p>Data——见 7.3.1 。</p>

7.4.9.2.7 ProxyActionThenGetResponseList 数据类型

响应代理操作后读取若干个服务器的若干个对象方法和属性（ProxyActionThenGetResponseList）的数据类型定义见表102 。

表 103 ProxyActionThenGetResponseList 数据类型定义

数据类型定义	说明
<pre>ProxyActionThenGetResponseList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 代理若干个服务器的操作后读取结果 SEQUENCE OF { 一个目标服务器地址 TSA, 若干个对象方法和属性操作后读取结果 SEQUENCE OF { 操作的对象方法描述符 OMD, 操作结果 DAR, 操作返回数据 Data OPTIONAL, 一个对象属性及结果 A-ResultNormal } } }</pre>	<p>PIID-ACD——见 0。</p> <p>TSA——见 7.3.10 。</p> <p>OMD——见 7.3.8 。</p> <p>DAR——见 7.3.22 。</p> <p>OAD——见 7.3.6 。</p> <p>Data——见 7.3.1 。</p> <p>A-ResultNormal——见表 53 。</p>

7.4.9.2.8 ProxyTransCommandResponse 数据类型

响应代理操作透明转发命令（ProxyTransCommandResponse）的数据类型定义见表103 。

表 104 ProxyTransCommandResponse 数据类型定义

数据类型定义	说明
<pre>ProxyTransCommandResponse ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 转发端口对象描述符 OAD, 转发命令返回结果 TransResult }</pre>	PIID-ACD——见 0。 OAD——见 7.3.6。
<pre>TransResult ::= CHOICE { 转发结果 [0] DAR, 返回数据 [1] octet-string }</pre>	DAR——见 7.3.22。

7.4.9.2.9 ProxyInnerTransCommandResponse 数据类型

响应代理终端内部透明转发命令（ProxyInnerTransCommandResponse）的数据类型定义见表103。

表 105 ProxyInnerTransCommandResponse 数据类型定义

数据类型定义	说明
<pre>ProxyInnerTransCommandResponse ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 转发命令返回结果 SEQUENCE OF InTransResult }</pre>	PIID-ACD——见 0。 OAD——见 7.3.6。
<pre>InTransResult ::= SEQUENCE { 逻辑地址 unsigned, 转发命令返回 CHOICE { 转发结果 [0] DAR, 返回数据 [1] octet-string } }</pre>	DAR——见 7.3.22。

7.4.10 安全传输

7.4.10.1 SECURITY-Request 数据类型

请求安全传输（SECURITY-Request）的数据类型定义见表104。

表 106 SECURITY-Request 数据类型定义

数据类型定义	说明
<pre> SECURITY-Request ::= SEQUENCE { 应用数据单元 CHOICE { 明文数据单元 [0] octet-string, 密文数据单元 [1] octet-string }, 数据验证信息 CHOICE { 数据验证码 [0] SID_MAC, 随机数 [1] RN, 随机数+MAC [2] RN_MAC, 安全标识 [3] SID } } </pre>	<p>SID——见 7.3.18 。</p> <p>RN——见 7.3.12 。</p> <p>SID_MAC——见 7.3.19 。</p>
<pre> RN_MAC ::= SEQUENCE { 随机数 RN, 消息鉴别码 MAC } </pre>	<p>MAC——见 7.3.11 。</p>

7.4.10.2 SECURITY-Response 数据类型

响应安全传输（SECURITY-Response）的数据类型定义见表105 。

表 107 SECURITY-Response 数据类型定义

数据类型定义	说明
<pre> SECURITY-Response ::= SEQUENCE { 应用数据单元 CHOICE { 明文数据单元 [0] octet-string, 密文数据单元 [1] octet-string, 安全传输错误码 [2] DAR }, 数据验证信息 CHOICE OPTIONAL { 数据MAC [0] MAC } } </pre>	<p>DAR——见 7.3.22 。</p> <p>MAC——见 7.3.11 。</p>

7.4.11 紧凑读取

7.4.11.1 COMPACT-GET-Request 数据类型

请求紧凑读取（COMPACT-GET-Request）的数据类型定义见表106。

表 108 COMPACT-GET-Request 数据类型定义

数据类型定义	说明
COMPACT-GET-Request ::= CHOICE { 请求紧凑读取一个对象属性 [1] GetRequestNormal, 请求紧凑读取多个对象属性 [2] GetRequestNormalList, 请求紧凑读取一个记录型对象属性 [3] GetRequestRecord }	

7.4.11.2 COMPACT-GET-Response 数据类型

7.4.11.2.1 概述

响应紧凑读取（COMPACT-GET-Response）的数据类型定义见表107。

表 109 COMPACT-GET-Response 数据类型定义

数据类型定义	说明
COMPACT-GET-Response ::= CHOICE { 紧凑读取一个对象属性的响应 [1] CompactGetResponseNormal, 紧凑读取多个对象属性的响应 [2] CompactGetResponseNormalList, 紧凑读取一个记录型对象属性的响应 [3] CompactGetResponseRecord }	

7.4.11.2.2 CompactGetResponseNormal 数据类型

响应紧凑读取一个对象属性（CompactGetResponseNormal）的数据类型定义见表108。

表 110 CompactGetResponseNormal 数据类型定义

数据类型定义	说明
CompactGetResponseNormal ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 一个对象属性及结果 Get-Result }	PIID-ACD——见 0。 Get-Result——见 7.4.5.2.2。

7.4.11.2.3 CompactGetResponseNormalList 数据类型

响应紧凑读取多个对象属性（CompactGetResponseNormalList）的数据类型定义见表109。

表 111 CompactGetResponseNormalList 数据类型定义

数据类型定义	说明
CompactGetResponseNormalList ::= SEQUENCE	PIID-ACD——见 0。

<pre>{ 服务序号-优先级-ACD PIID-ACD, 多个对象属性及结果 SEQUENCE OF Get-Result }</pre>	Get-Result——见 7.4.5.2.2 。
--	---------------------------

7.4.11.2.4 CompactGetResponseRecord 数据类型

响应紧凑读取一个记录型对象属性（CompactGetResponseRecord）的数据类型定义见表110 。

表 112 CompactGetResponseRecord 数据类型定义

数据类型定义	说明
<pre>CompactGetResponseRecord ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 一个记录型对象属性及结果 A-CompactResultRecord }</pre>	PIID-ACD——见 0。
<pre>A-CompactResultRecord ::= CHOICE { 错误信息 [0] DAR, M 条记录 [1] SEQUENCE OF A-RecordRow }</pre>	A-RecordRow——见 7.4.5.2.2 。

7.4.12 紧凑设置

7.4.12.1 COMPACT-SET-Request 数据类型

请求紧凑设置（COMPACT-SET-Request）的数据类型定义见表111 。

表 113 COMPACT-SET-Request 数据类型定义

数据类型定义	说明
<pre>COMPACT-SET-Request ::= CHOICE { 请求紧凑设置一个对象属性 [1] SetRequestNormal, 请求紧凑设置多个对象属性 [2] SetRequestNormalList }</pre>	

7.4.12.2 COMPACT-SET-Response 数据类型

7.4.12.2.1 概述

响应紧凑设置（COMPACT-SET-Response）的数据类型定义见表112 。

表 114 COMPACT-SET-Response 数据类型定义

数据类型定义	说明
--------	----

COMPACT-SET-Response ::= CHOICE { 紧凑设置一个对象属性的响应 [1] CompactSetResponseNormal, 紧凑设置多个对象属性的响应 [2] CompactSetResponseNormalList }	
--	--

7.4.12.2.2 CompactSetResponseNormal 数据类型

响应紧凑设置一个对象属性（CompactSetResponseNormal）的数据类型定义见表113。

表 115 CompactSetResponseNormal 数据类型定义

数据类型定义	说明
SetResponseNormal ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 设置执行结果 DAR }	PIID-ACD——见 0。 DAR——见 7.3.22。

7.4.12.2.3 CompactSetResponseNormalList 数据类型

响应紧凑设置多个对象属性（CompactSetResponseNormalList）的数据类型定义见表114。

表 116 CompactSetResponseNormalList 数据类型定义

数据类型定义	说明
SetResponseNormalList ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 设置执行结果 SEQUENCE OF DAR }	PIID-ACD——见 0。 DAR——见 7.3.22。

7.4.13 紧凑代理

7.4.13.1 COMPACT-PROXY-Request 数据类型

7.4.13.1.1 概述

请求紧凑代理（COMPACT-PROXY-Request）的数据类型定义见表115。

表 117 COMPACT-PROXY-Request 数据类型定义

数据类型定义	说明
COMPACT-PROXY-Request ::= CHOICE { 请求紧凑代理读取服务器的若干个对象属性 [1] CompactProxyGetRequestNormal,	

请求紧凑代理读取服务器的一个记录型对象属性	[2] CompactProxyGetRequestRecord,	
请求紧凑代理设置服务器的若干个对象属性	[3] CompactProxySetRequestNormal,	
请求紧凑代理设置后读取服务器的若干个对象属性	[4] CompactProxySetThenGetRequest,	
请求紧凑代理操作服务器的若干个对象方法	[5] CompactProxyActionRequest,	
请求紧凑代理操作后读取服务器的若干个对象方法和属性	[6] CompactProxyActionThenGetRequest,	
}		

7.4.13.1.2 CompactProxyGetRequestNormal 数据类型

请求紧凑代理读取服务器的若干个对象属性 (CompactProxyGetRequestNormal) 的数据类型定义见表116。

表 118 CompactProxyGetRequestNormal 数据类型定义

数据类型定义	说明
CompactProxyGetRequestNormal ::= SEQUENCE { 服务序号-优先级 PIID, 代理整个请求的超时时间 long-unsigned, 目标服务器地址 TSA, 若干个对象属性描述符 SEQUENCE OF OAD }	PIID——见 7.3.3。 TSA——见 7.3.10。 OAD——见 7.3.6。 代理整个请求的超时时间——单位：秒，非 0 值。

7.4.13.1.3 CompactProxyGetRequestRecord 数据类型

请求紧凑代理读取服务器的一个记录型对象属性 (CompactProxyGetRequestRecord) 的数据类型定义见表117。

表 119 CompactProxyGetRequestRecord 数据类型定义

数据类型定义	说明
CompactProxyGetRequestRecord ::= SEQUENCE { 服务序号-优先级 PIID, 代理请求的超时时间 long-unsigned, 目标服务器地址 TSA, 对象属性描述符 OAD, 记录行选择描述符 RSD, 记录列选择描述符 RCSD }	PIID——见 7.3.3。 TSA——见 7.3.10。 OAD——见 7.3.6。 RSD——见 0。 RCSD——见 7.3.21。 代理请求的超时时间——单位：秒，非 0 值。

7.4.13.1.4 CompactProxySetRequest 数据类型

请求紧凑代理设置服务器的若干个对象属性 (CompactProxySetRequest) 数据类型定义见表118。

表 120 CompactProxySetRequestList 数据类型定义

数据类型定义	说明
<pre>CompactProxySetRequestList ::= SEQUENCE { 服务序号-优先级 PIID, 代理整个请求的超时时间 long-unsigned, 目标服务器地址 TSA, 若干个对象属性描述符及数据 SEQUENCE OF { 对象属性描述符 OAD, 设置数值 Data } }</pre>	PIID——见 7.3.3 。 TSA——见 7.3.10 。 OAD——见 7.3.6 。 Data——见 7.3.1 。 代理整个请求的超时时间——单位：秒，非 0 值。

7.4.13.1.5 CompactProxySetThenGetRequest 数据类型

请求紧凑代理设置后读取服务器的若干个对象属性（CompactProxySetThenGetRequest）的数据类型定义见表119。

表 121 CompactProxySetThenGetRequest 数据类型定义

数据类型定义	说明
<pre>CompactProxySetThenGetRequest ::= SEQUENCE { 服务序号-优先级 PIID, 代理整个请求的超时时间 long-unsigned, 目标服务器地址 TSA, 若干个对象属性的设置后读取 SEQUENCE OF { 设置的对象属性描述符 OAD, 设置数值 Data, 读取的对象属性描述符 OAD, 延时读取时间 unsigned } }</pre>	PIID——见 7.3.3 。 TSA——见 7.3.10 。 OAD——见 7.3.6 。 Data——见 7.3.1 。 代理整个请求的超时时间——单位：秒，非 0 值。 延时读取时间——单位：秒，0 表示由目标服务器确定延时读取时间。

7.4.13.1.6 CompactProxyActionRequest 数据类型

请求精简代理操作服务器的若干个对象方法（CompactProxyActionRequest）的数据类型定义见表 120。

表 122 CompactProxyActionRequest 数据类型定义

数据类型定义	说明
<pre>CompactProxyActionRequest ::= SEQUENCE { 服务序号-优先级 PIID,</pre>	PIID——见 7.3.3 。 TSA——见 7.3.10 。 OMD——见 7.3.8 。

数据类型定义	说明
代理整个请求的超时时间 long-unsigned, 目标服务器地址 TSA, 若干个对象方法描述符及参数 SEQUENCE OF { 对象方法描述符 OMD, 方法参数 Data } }	Data——见 7.3.1 。 代理整个请求的超时时间——单位：秒，非 0 值。

7.4.13.1.7 CompactProxyActionThenGetRequest 数据类型

请求精简代理操作后读取服务器的若干个对象方法和属性（CompactProxyActionThenGetRequest）的数据类型定义见表121。

表 123 CompactProxyActionThenGetRequest 数据类型定义

数据类型定义	说明
CompactProxyActionThenGetRequest ::= SEQUENCE { 服务序号-优先级 PIID, 代理整个请求的超时时间 long-unsigned, 目标服务器地址 TSA, 若干个对象方法及属性的操作后读取 SEQUENCE OF { 操作的对象方法描述符 OMD, 方法参数 Data, 读取的对象属性描述符 OAD, 延时读取时间 unsigned } }	PIID——见 7.3.3 。 TSA——见 7.3.10 。 OMD——见 7.3.8 。 Data——见 7.3.1 。 OAD——见 7.3.6 。 代理整个请求的超时时间——单位：秒，非 0 值。 延时读取时间——单位：秒，0 表示由目标服务器确定延时读取时间。

7.4.13.2 COMPACT-PROXY-Response 数据类型

7.4.13.2.1 概述

响应精简代理（COMPACT-PROXY-Response）的数据类型定义见表122。

表 124 COMPACT-PROXY-Response 数据类型定义

数据类型定义	说明
--------	----

数据类型定义	说明
COMPACT-PROXY-Response ::= CHOICE { 代理精简读取服务器的若干个对象属性的响应 [1] CompactProxyGetResponseNormal, 代理精简读取服务器的一个记录型对象属性的响应 [2] CompactProxyGetResponseRecord, 代理精简设置服务器的若干个对象属性的响应 [3] CompactProxySetResponse, 代理精简设置后读取服务器的若干个对象属性的响应 [4] CompactProxySetThenGetResponse, 代理精简操作服务器的若干个对象方法的响应 [5] CompactProxyActionResponse, 代理精简操作后读取服务器的若干个对象方法和属性的响应 [6] CompactProxyActionThenGetResponse }	

7.4.13.2.2 CompactProxyGetResponseNormal 数据类型

响应精简代理读取服务器的若干个对象属性（CompactProxyGetResponseNormal）的数据类型定义见表123。

表 125 ProxyGetResponseList 数据类型定义

数据类型定义	说明
CompactProxyGetResponseNormal ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 若干个对象属性及结果 SEQUENCE OF Get-Result }	PIID-ACD——见 0。 Get-Result——见 7.4.5.2.2。

7.4.13.2.3 CompactProxyGetResponseRecord 数据类型

响应代理读取服务器的一个记录型对象属性（CompactProxyGetResponseRecord）的数据类型定义见表124。

表 126 ProxyGetResponseRecord 数据类型定义

数据类型定义	说明
ProxyGetResponseRecord ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 一个精简记录型对象属性及结果 A-CompactResultRecord }	PIID-ACD——见 0。 A-CompactResultRecord——见 7.4.11.2.4。

7.4.13.2.4 CompactProxySetResponse 数据类型

响应精简代理设置服务器的若干个对象属性（CompactProxySetResponse）的数据类型定义见表125。

表 127 CompactProxySetResponse 数据类型定义

数据类型定义	说明

数据类型定义	说明
CompactProxySetResponse ::= SEQUENCE { 服务序号-优先级-ACD PIIID-ACD, 若干个对象属性描述符及结果 SEQUENCE OF DAR }	PIIID-ACD——见 0。 DAR——见 7.3.22 。

7.4.13.2.5 CompactProxySetThenGetResponse 数据类型

响应精简代理设置后读取服务器的若干个对象属性 (CompactProxySetThenGetResponse) 的数据类型定义见表 126 。

表 128 CompactProxySetThenGetResponse 数据类型定义

数据类型定义	说明
CompactProxySetThenGetResponse ::= SEQUENCE { 服务序号-优先级-ACD PIIID-ACD, 若干个对象属性设置后读取结果 SEQUENCE OF { 设置结果 DAR, 一个对象属性及结果 Get-Result } }	PIIID-ACD——见 0。 DAR——见 7.3.22 。 Get-Result——见 7.4.5.2.2 。

7.4.13.2.6 CompactProxyActionResponse 数据类型

响应精简代理操作服务器的若干个对象方法 (CompactProxyActionResponse) 的数据类型定义见表 127 。

表 129 CompactProxyActionResponse 数据类型定义

数据类型定义	说明
CompactProxyActionResponse ::= SEQUENCE { 服务序号-优先级-ACD PIIID-ACD, 若干个对象方法描述符及结果 SEQUENCE OF { 操作结果 DAR, 操作返回数据 Data OPTIONAL } }	PIIID-ACD——见 0。 DAR——见 7.3.22 。 Data——见 7.3.1 。

7.4.13.2.7 CompactProxyActionThenGetResponse 数据类型

响应精简代理操作后读取服务器的若干个对象方法和属性(CompactProxyActionThenGetResponse)的数据类型定义见表128。

表 130 CompactProxyActionThenGetResponse 数据类型定义

数据类型定义	说明
CompactProxyActionThenGetResponse ::= SEQUENCE { 服务序号-优先级-ACD PIID-ACD, 若干个对象方法和属性操作后读取结果 SEQUENCE OF { 操作结果 DAR, 操作返回数据 Data OPTIONAL, 一个对象属性及结果 Get-Result } }	PIID-ACD——见 0。 DAR——见 7.3.22。 Data——见 7.3.1。 Get-Result——见 7.4.5.2.2。

7.4.14 跟随上报信息域

跟随上报信息域 (FollowReport) 的数据类型定义见表129。

表 131 FollowReport 数据类型定义

数据类型定义	说明
FollowReport ::= CHOICE { 若干个对象属性及数据 [1] SEQUENCE OF A-ResultNormal, 若干个记录型对象属性及数据 [2] SEQUENCE OF A-ResultRecord }	A-ResultNormal——见表 53。 A-ResultRecord——见表 55。

7.4.15 时间标签域

时间标签域 (TimeTag) 的数据类型定义见表130。

表 132 TimeTag 数据类型定义

数据类型定义	说明
TimeTag ::= SEQUENCE { 发送时标 date_time_s, 允许传输延时时间 TI }	date_time_s——见 0。 TI——见 7.3.9。

7.4.16 异常响应

7.4.16.1 ERROR-Response 数据类型

响应异常 (ERROR-Response) 的数据类型定义见表131。

表 133 ERROR-Response 数据类型定义

数据类型定义	说明
<pre> ERROR-Response ::= SEQUENCE { 服务序号-优先级 PIID, 异常类型 ENUMERATED { APDU 无法解析 (1), 服务不支持 (2), 其他 (255) } } </pre>	PIID——见 7.3.3 。

8 接口类与对象标识

8.1 对象模型

对象是属性和方法的集合。对象的信息包含在属性中，属性值表示对象的特征。对象提供了操作的方法。具有共享公共特征的对象归纳为接口类（IC），接口类由类标识码（class_id）进行标识。接口类的公共特征（包括属性和方法）适用于该类的所有实例。接口类的实例称为接口类对象，简称对象。一个对象只能属于一个接口类。一个对象对应于一个唯一的标识，即对象标识（OI）。

8.2 接口类

8.2.1 接口类的描述方法

接口类在本标准中采用表格形式描述，其中包含类名、属性、方法等，见表132。

表 134 接口类表述格式

类名, class_id=n		实例数
属性		数据类型
1. 逻辑名	(static)	octet-string
2.	(.....)
3.	(.....)
方法		必选/可选
1.
2.

其中：

- 类名：类的名称，如：电能量类、最大需量类、冻结类等；
- 类标识码：即 class_id, 数值范围 0...255；200...255 用作制造商专属接口类；
- 属性：定义类的属性，静态（static）表示服务器不能自行更改，例如：配置参数；动态（dyn.）表示具有过程特性的属性，由服务器自行刷新。属性的编号即对象属性描述符（OAD）的属性标识；
- 逻辑名：对象标识（OI），标记接口类对象实例，见 8.2.29；
- 方法：定义类的方法，方法的编号即对象方法描述符（OMD）的方法标识；
- 实例数：定义逻辑设备内实例化对象的数量范围；
- 数据类型：定义属性的数据类型。

8.2.2 电能量类

本接口类定义了电能量数据信息，定义见表133。

表 135 电能量接口类定义

电能量类, class_id=1		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 总及费率电能量数组	(dyn.)	array

电能量类, class_id=1		0...n
3. 换算及单位	(static)	Scaler_Unit
4. 扩展精度总及费率电能量数组	(dyn.)	array
5. 扩展精度换算及单位	(static)	Scaler_Unit
方法		必选/可选
1. 复位		可选
2. 执行		可选

电能量类属性说明见表134。

表 136 电能量类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	总及费率电能量数组::=array 电能量 电能量: CHOICE { double-long-unsigned [6], double-long [5] }	总及 n 个费率的电能量, 第一个数组元素是总电能量, 从第二个数组元素依次排列费率 1...n 电能量。
3	换算及单位::=Scaler_Unit	见 7.3.14。
4	扩展精度总及费率电能量数组::=array 扩展精度电能量 扩展精度电能量: CHOICE { long64-unsigned [21], long64 [20] }	总及 n 个费率的电能量, 第一个数组元素是总电能量, 从第二个数组元素依次排列费率 1...n 电能量。
5	扩展精度换算及单位::=Scaler_Unit	见 7.3.14。

电能量类方法说明见表135。

表 137 电能量类方法说明

编号	方法	说明
1	复位 (参数) 参数::=integer (0)	通用方法, instance-specific。
2	执行 (参数) 参数::=Data	通用方法, instance-specific。

8.2.3 最大需量类

本接口类定义了最大需量数据信息, 定义见表136。

表 138 最大需量类定义

最大需量类, class_id = 2		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 总及费率最大需量数组	(dyn.)	array
3. 换算及单位	(static)	Scaler_Unit
方法		必选/可选
1. 复位		可选
2. 执行		可选

最大需量类属性说明见表137。

表 139 最大需量类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	总及费率最大需量数组::=array 最大需量及发生时间 最大需量及发生时间::=structure { 最大需量值 CHOICE, 发生时间 date_time_s } 最大需量值: CHOICE { double-long [5], double-long-unsigned [6] }	总及 n 个费率的 <code>最大需量</code> ，第一个数组元素是总 <code>最大需量</code> ，从第二个数组元素依次排列费率 1...n <code>最大需量</code> 。
3	换算及单位::=Scaler_Unit	见 7.3.14。 最大需量发生时间无换算单位。

最大需量类方法说明见表138。

表 140 最大需量类方法说明

编号	方法	说明
1	复位 (参数) 参数::=integer (0)	通用方法, instance-specific。
2	执行 (参数) 参数::=Data	通用方法, instance-specific。

8.2.4 分相变量类

本接口类定义了电压、电流、相角等分相变量数据信息，定义见表139。

表 141 分相变量类定义

分相变量类, class_id=3		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 分相数值数组	(dyn.)	array
3. 换算及单位	(static)	Scaler_Unit
方法		必选/可选
1. 复位		可选
2. 执行		可选

分相变量类属性说明见表140。

表 142 分相变量类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	分相数值数组::=array 分相数值 分相数值::=instance-specific	数值数组按 A 相、B 相、C 相顺序排列，单相时，数值数组仅包含一个元素。
3	换算及单位::=Scaler_Unit	见 7.3.14。

分相变量类方法说明见表141。

表 143 分相变量类方法说明

编号	方法	说明
1	复位（参数） 参数::=integer (0)	通用方法，instance-specific。
2	执行（参数） 参数::=Data	通用方法，instance-specific。

8.2.5 功率类

本接口类定义了功率、功率因数等数据信息，定义见表142。

表 144 功率类定义

功率类, class_id=4		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 总及分相数值数组	(dyn.)	array
3. 换算及单位	(static)	Scaler_Unit
方法		必选/可选
1. 复位		可选
2. 执行		可选

功率类属性说明见表143。

表 145 功率类属性说明

编号	属性	说明
1	逻辑名:: <code>octet-string</code>	
2	总及分相数值数组:: <code>array</code> 数值 数值:: <code>instance-specific</code>	数值数组按总、A相、B相、C相顺序排列，单相时，数值数组包含两个元素，为总及单相数值。
3	换算及单位:: <code>Scaler_Unit</code>	见 7.3.14。

功率类方法说明见表144。

表 146 功率类方法说明

编号	方法	说明
1	复位 (参数) 参数:: <code>integer (0)</code>	通用方法， <code>instance-specific</code> 。
2	执行 (参数) 参数:: <code>Data</code>	通用方法， <code>instance-specific</code> 。

8.2.6 谐波变量类

本接口类定义了谐波变量数据信息，定义见表145。

表 147 谐波变量类定义

谐波变量类, <code>class_id=5</code>		0...n
属性		数据类型
1. 逻辑名	(static)	<code>octet-string</code>
2. A相各次谐波数值数组	(dyn.)	<code>array</code>
3. B相各次谐波数值数组	(dyn.)	<code>array</code>
4. C相各次谐波数值数组	(dyn.)	<code>array</code>
5. 最高谐波次数	(static)	<code>unsigned</code>
6. 换算及单位	(static)	<code>Scaler_Unit</code>
方法		必选/可选
1. 复位		可选
2. 执行		可选

谐波变量类属性说明见表146。

表 148 谐波变量类属性说明

编号	属性	说明
1	逻辑名:: <code>octet-string</code>	
2	A相各次谐波数值数组:: <code>array</code> 数值 数值:: <code>instance-specific</code>	A相各次谐波相关数值。
3	B相各次谐波数值数组:: <code>array</code> 数值 数值:: <code>instance-specific</code>	B相各次谐波相关数值。

编号	属性	说明
4	C相各次谐波数值组::=array 数值 数值::=instance-specific	C相各次谐波相关数值。
5	最高谐波次数	表示限定谐波相关数值数组中的最高谐波次数。
6	换算及单位::=Scaler_Unit	见 7.3.14 。

谐波变量类方法说明见表147 。

表 149 谐波变量类方法说明

编号	方法	说明
1	复位（参数） 参数::=integer (0)	通用方法，instance-specific。
2	执行（参数） 参数::=Data	通用方法，instance-specific。

8.2.7 数据变量类

本接口类定义了过程值或与过程值单元相关的状态值数据信息，定义见表148 。

表 150 数据变量类定义

数据变量类，class_id=6		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 数值	(dyn.)	instance-specific
3. 换算及单位	(static)	Scaler_Unit
方法		必选/可选
1. 复位		可选
2. 执行		可选

数据变量类属性说明见表149 。

表 151 数据变量类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	数值::=instance-specific	过程值或与过程值单元相关的状态值，数据类型由“逻辑名”引用的对象实例定义。
3	换算及单位::=Scaler_Unit	见 7.3.14 。

数据变量类方法说明见表150 。

表 152 数据变量类方法说明

编号	方法	说明
1	复位（参数）	通用方法，instance-specific。

	参数::=integer (0)	
2	执行 (参数) 参数::=Data	通用方法, instance-specific。

8.2.8 事件对象类

本接口类定义了配置、存储事件记录数据信息，定义见表151。

表 153 事件对象类定义

事件对象类, class_id=7		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 事件记录表	(dyn.)	array
3. 关联对象属性表	(static)	array
4. 当前记录数	(dyn.)	long-unsigned
5. 最大记录数	(static)	long-unsigned
6. 配置参数	(static)	structure
7. 当前值记录表	(dyn.)	array
8. 上报标识	(static)	enum
9. 有效标识	(static)	bool
10. 时间状态记录表	(static)	array
方法		必选/可选
1. 复位		可选
2. 执行		可选
4. 添加一个事件关联对象属性		可选
5. 删除一个事件关联对象属性		可选

事件对象类属性说明见表152。

表 154 事件对象类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	事件记录表::=array 一条事件记录 一条事件记录::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 instance-specific, 事件上报状态 array 通道上报状态, 事件特殊数据 1 instance-specific, ... 事件特殊数据 N instance-specific, 第 1 个关联对象属性的数据 Data, ... }	用于存储事件记录，记录根据“配置参数”自动产生。 记录的顺序按照事件发生的次序排序。执行“复位”方法后事件记录表清空。 事件记录序号——单调递增。 事件发生源——由“逻辑名”引用的对象实例定义。 事件上报状态——按通道分别记录上报状态。 事件特殊数据——由“逻辑名”引用的对象实例定义。 关联对象属性的数据——其排列次序和数量 n，由“逻辑名”引用的对象实例的属性 3 定义。

编号	属性	说明
	第 n 个关联对象属性的数据 Data } 通道上报状态::=structure { 通道 OAD, 上报状态 unsigned } 上报状态: bit0:事件发生上报标识, 0—未上报, 1—已上报; bit1:事件发生上报确认标识, 0—未确认, 1—已确认; bit2:事件结束(恢复)上报标识, 0—未上报, 1—已上报; bit3:事件结束(恢复)上报确认标识, 0—未确认, 1—已确认。	
3	关联对象属性表::=array 对象属性 对象属性::=OAD OAD的“属性特征”取值如下: 1: 事件发生前; 2: 事件发生后; 3: 事件结束前; 4: 事件结束后。	事件记录中的关联对象属性数据依据关联对象属性表产生。
4	当前记录数::=long-unsigned	表示保存在事件记录表中的记录数。执行“复位”方法后, 记录表清空, 此时当前记录数为0。
5	最大记录数::=long-unsigned	事件记录表最小存储条数。
6	配置参数::=structure { 参数 1 instance-specific, ... 参数 n instance-specific }	配置触发事件记录的判定参数, 参数的数据类型由“逻辑名”引用的对象实例定义, 见7.3.5。
7	当前值记录表::=array 当前值 当前值::=structure { 事件发生源 instance-specific, 事件发生时间统计 structure { 事件发生次数 double-long-unsigned, 事件累计时间 double-long-unsigned } }	事件发生源, 由“逻辑名”引用的对象实例定义; 当前值包含事件发生的次数以及累计时间; 事件发生次数——单位: 次; 事件累计时间——单位: 秒。
8	上报标识::=enum {	

编号	属性	说明
	不上报 (0), 事件发生上报 (1), 事件恢复上报 (2), 事件发生恢复均上报 (3) }	
9	有效标识::=bool	
10	时间状态记录表::=array 时间状态 时间状态::=structure { 事件发生源 instance-specific, 最近一次时间 structure { 最近一次发生时间 date_time_s, 最近一次结束时间 date_time_s } }	最近一次发生时间——上 1 次事件的发生时间，如果未发生返回 NULL。 最近一次结束时间——上 1 次事件的结束时间，如上 1 次事件未结束，则取上 2 次事件的结束时间；不存在上 2 次事件或结束时间不支持，返回 NULL。

事件对象类方法说明见表153。

表 155 事件对象类方法说明

编号	方法	说明
1	复位 (参数) 参数::=integer (0)	通用方法，instance-specific。
2	执行 (参数) 参数::=Data	通用方法，instance-specific。
4	添加一个事件关联对象属性 (参数) 参数::=OAD 对象属性描述符	向属性“关联对象属性表”中，增加一个关联对象属性。 参数——对象属性描述符。
5	删除一个事件关联对象属性 (参数) 参数::=OAD 对象属性描述符	从属性“关联对象属性表”中，删除一个关联对象属性。 参数——对象属性描述符。

8.2.9 参数变量类

本接口类定义了参数信息，定义见表154。

表 156 参数变量类定义

参数变量类，class_id=8		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 参数	(static)	instance-specific
方法		必选/可选
1. 复位		可选
2. 执行		可选

参数变量类属性说明见表155。

表 157 参数变量类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	参数::=instance-specific	参数的数据类型由“逻辑名”引用的对象实例定义。

参数变量类方法说明见表156。

表 158 参数变量类方法说明

编号	方法	说明
1	复位（参数） 参数::=integer (0)	通用方法，instance-specific。
2	执行（参数） 参数::=Data	通用方法，instance-specific。

8.2.10 冻结数据类

本接口类定义了配置、存储冻结数据及相关信息，定义见表157。

表 159 冻结数据类定义

冻结数据类，class_id=9		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 冻结数据表	(dyn.)	array
3. 关联对象属性表	(static)	array
方法		可选/可选
1. 复位		可选
2. 执行		可选
3. 触发一次冻结		可选
4. 添加一个冻结对象属性		可选
5. 删除一个冻结对象属性		可选
7. 批量添加冻结对象属性		可选
8. 清除关联对象属性表		可选

冻结数据类属性说明见表158。

表 160 冻结数据类属性说明

编号	属性	说明
1	逻辑名::=octet-string	

编号	属性	说明
2	冻结数据表::=array 一条冻结记录 一条冻结记录::=structure { 冻结记录序号 double-long-unsigned, 冻结时间 date_time_s, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }	冻结记录可根据“冻结周期”和“延时时间”自动冻结产生或执行“触发一次冻结”产生。 冻结记录按照冻结发生的次序排列。复位后冻结数据表清空。 冻结记录序号——单调递增。 关联对象属性的数据——其排列次序和数量 n, 由“逻辑名”引用的对象实例的属性 3 定义。
3	关联对象属性表::=array 一个关联的对象属性 一个关联的对象属性::=structure { 冻结周期 long-unsigned, 关联对象属性描述符 OAD, 存储深度 long-unsigned } 关联对象属性描述符: 对象属性描述符的属性特征取值 0...7, 关联对象属性描述符在关联对象属性表中具有唯一性。	冻结记录中的关联对象属性数据依据关联对象属性表产生。 冻结周期——冻结操作的间隔时间, 其数值单位由“逻辑名”引用的对象实例定义。当冻结周期为 0 时: 表示由外部条件或异步发生的冻结事件触发 (由执行“触发一次冻结”) 产生。 关联对象属性描述符: 对象属性描述符的属性特征, 每一种属性特征表示一种冻结方案, 每种方案可分配不同的冻结周期以及存储深度; 存储深度: 关联对象属性数据冻结存储的最少记录条数。
4	配置参数::= instance-specific	具体配置参由“逻辑名”引用的对象实例定义。

冻结数据类方法说明见表159。

表 161 冻结数据类方法说明

编号	方法	说明
1	复位 (参数) 参数::=integer (0)	通用方法, instance-specific。
2	执行 (参数) 参数::=Data	通用方法, instance-specific。
3	触发一次冻结 (参数) 参数::=long-unsigned 延时执行的时间	触发一次数据冻结操作。 参数——延时执行的时间, 单位: 秒, 0 表示立即执行。
4	添加一个冻结对象属性 (冻结对象) 冻结对象::=structure { 冻结周期 long-unsigned, 关联对象属性描述符 OAD, 存储深度 long-unsigned }	向属性“关联对象属性表”中, 增加一个冻结对象。
5	删除一个冻结对象属性 (参数) 参数::=OAD 对象属性描述符	从属性“关联对象属性表”中, 删除一个冻结对象。

编号	方法	说明
7	批量添加冻结对象属性 (array 冻结对象)	冻结对象定义见方法 4。
8	清除关联对象属性表 (参数) 参数:: <code>=NULL</code>	

8.2.11 采集类

本接口类定义了配置和存储与采集相关的参数、数据和记录。定义见表160。

表 162 采集类定义

采集类, class_id=10		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 配置表	(static)	array
3. 记录表	(dyn.)	array
方法		必选/可选
1. 复位		可选
2. 执行		可选
3. 清空记录表		可选

采集类属性说明见表161。

表 163 采集类属性说明

编号	属性	说明
1	逻辑名:: <code>=octet-string</code>	
2	配置表:: <code>=array</code> 配置单元 配置单元:: <code>=instance-specific</code>	配置单元——配置采集相关参数，其内容及数据类型由“逻辑名”引用的对象实例定义。见 A.8。
3	记录表:: <code>=array</code> 记录单元 记录单元:: <code>=instance-specific</code>	记录单元——记录采集相关的数据，其内容及数据类型由“逻辑名”引用的对象实例定义。见 A.8。

采集类方法说明见表162。

表 164 采集类方法说明

编号	方法	说明
1	复位 (参数) 参数:: <code>=integer (0)</code>	通用方法, instance-specific。
2	执行 (参数) 参数:: <code>=Data</code>	通用方法, instance-specific。
3	清空记录表 (参数) 参数:: <code>=NULL</code>	

8.2.12 集合类

本接口类定义了一种通用的资料信息集合，定义见表163。

表 165 集合类定义

集合类, class_id=11		0…n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 集合	(dyn.)	array
3. 当前元素个数	(dyn.)	long-unsigned
4. 最大元素个数	(static)	long-unsigned
方法		必选/可选
1. 复位		可选
2. 执行		可选

集合类属性说明见表164。

表 166 集合类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	集合::=array 集合元素 集合元素::=instance-specific	集合元素——其内容及数据类型由“逻辑名”引用的对象实例定义。
3	当前元素个数	表示保存在记录表中的记录数。执行“复位”方法后，记录表清空，当前记录数为0。
4	最大元素个数	

集合类方法说明见表165。

表 167 集合类方法说明

编号	方法	说明
1	复位(参数) 参数::=integer(0)	通用方法, instance-specific。
2	执行(参数) 参数::=Data	通用方法, instance-specific。

8.2.13 脉冲计量类

本接口类定义了脉冲计量相关配置和数据信息，定义见表166。

表 168 脉冲计量类定义

脉冲计量类, class_id=12		0…n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 通信地址	(static)	octet-string
3. 互感器倍率	(static)	structure
4. 脉冲配置	(static)	array
5. 有功功率	(dyn.)	double-long

脉冲计量类, class_id=12		0…n
6. 无功功率	(dyn.)	double-long
7. 当日正向有功电量	(dyn.)	array
8. 当月正向有功电量	(dyn.)	array
9. 当日反向有功电量	(dyn.)	array
10. 当月反向有功电量	(dyn.)	array
11. 当日正向无功电量	(dyn.)	array
12. 当月正向无功电量	(dyn.)	array
13. 当日反向无功电量	(dyn.)	array
14. 当月反向无功电量	(dyn.)	array
15. 正向有功电能示值	(dyn.)	array
16. 正向无功电能示值	(dyn.)	array
17. 反向有功电能示值	(dyn.)	array
18. 反向无功电能示值	(dyn.)	array
19. 换算及单位	(static)	structure
方法		必选/可选
1. 复位		可选
2. 执行		可选
3. 添加脉冲输入单元		必选
4. 删除脉冲输入单元		必选

脉冲计量类的属性描述见表167。

表 169 脉冲计量类属性描述

编号	属性	说明
1	逻辑名::=octet-string	
2	通信地址::=octet-string	
3	互感器倍率::=structure { PT long-unsigned, CT long-unsigned }	
4	脉冲配置::=array 脉冲单元 脉冲单元::=structure { 脉冲输入端口 0AD, 脉冲属性 enum { 正向有功 (0), 正向无功 (1), 反向有功 (2), 反向无功 (3) } }	

编号	属性	说明
	<pre> }, 脉冲常数 k long-unsigned } </pre>	
5	有功功率::=double-long	
6	无功功率::=double-long	
7	当日正向有功电量::=array 电能量 电能量::=double-long-unsigned	总及 n 个费率的电能量，第一个数组元素是总电能量，从第二个数组元素依次排列费率 1…n 电能量。
8	当月正向有功电量::=array 电能量 电能量::=double-long-unsigned	同上。
9	当日反向有功电量::=array 电能量 电能量::=double-long-unsigned	同上。
10	当月反向有功电量::=array 电能量 电能量::=double-long-unsigned	同上。
11	当日正向无功电量::=array 电能量 电能量::=double-long-unsigned	同上。
12	当月正向无功电量::=array 电能量 电能量::=double-long-unsigned	同上。
13	当日反向无功电量::=array 电能量 电能量::=double-long-unsigned	同上。
14	当月反向无功电量::=array 电能量 电能量::=double-long-unsigned	同上。
15	正向有功电能示值::=array 电能示值 电能示值::=double-long-unsigned	同上。
16	正向无功电能示值::=array 电能示值 电能示值::=double-long-unsigned	同上。
17	反向有功电能示值::=array 电能示值 电能示值::=double-long-unsigned	同上。
18	反向无功电能示值::=array 电能示值 电能示值::=double-long-unsigned	同上。
19	单位及换算::=structure <pre> { 属性 5 单位及换算 Scaler_Unit (单位: W, 换算: -1), 属性 6 单位及换算 Scaler_Unit (单位: var, 换算: -1), 属性 7 单位及换算 Scaler_Unit (单位: kWh, 换算: -4), 属性 8 单位及换算 Scaler_Unit (单位: kWh, 换算: -4), 属性 9 单位及换算 Scaler_Unit (单位: kWh, 换算: -4), 属性 10 单位及换算 Scaler_Unit (单位: kWh, 换算: -4), 属性 11 单位及换算 Scaler_Unit (单位: kvarh, 换算: -4), 属性 12 单位及换算 Scaler_Unit (单位: kvarh, 换算: -4), 属性 13 单位及换算 Scaler_Unit (单位: kvarh, 换算: -4), </pre>	Scaler_Unit 见表 17。

编号	属性	说明
	属性 14 单位及换算 Scaler_Unit(单位: kvarh, 换算: -4), 属性 15 单位及换算 Scaler_Unit(单位: kWh, 换算: -4), 属性 16 单位及换算 Scaler_Unit(单位: kWh, 换算: -4), 属性 17 单位及换算 Scaler_Unit(单位: kvarh, 换算: -4), 属性 18 单位及换算 Scaler_Unit(单位: kvarh, 换算: -4) }	

脉冲计量类的方法描述见表 168 。

表 170 脉冲计量类方法描述

编号	方法	说明
1	复位(参数) 参数::=bit-string	通用方法, instance-specific。
2	执行(参数) 参数::=Data	通用方法, instance-specific。
3	添加脉冲输入单元(脉冲单元)	
4	删除脉冲输入单元(脉冲输入端口)	

8.2.14 控制类

本接口类定义了用电负荷控制的功能, 定义见表169 。

表 171 控制类定义

控制类, class_id=13		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 控制方案集	(static)	array
3. 控制投入状态	(dyn.)	array
4. 控制输出状态	(dyn.)	array
5. 越限告警状态	(dyn.)	array
方法		必选/可选
1. 复位		可选
2. 执行		可选
3. 添加控制方案单元		必选
4. 删除控制方案单元		必选
5. 更新控制方案单元		必选
6. 控制投入		必选
7. 控制解除		必选

控制类属性说明见表170 。

表 172 控制类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	控制方案集::=array 控制方案单元 控制方案单元::=instance-specific	控制方案单元——其内容及数据类型由“逻辑名”引用的对象实例定义。
3	控制投入状态::=array 一个总加组控制投入状态 一个总加组控制投入状态::=structure { 总加组对象 OI, 投入状态 enum{未投入(0), 投入(1)} }	
4	控制输出状态::=array 一个总加组控制输出状态 一个总加组控制输出状态::=structure { 总加组对象 OI, 控制输出状态 bit-string(SIZE(8)) } 控制输出状态: bit0…bit7 对应第1…第8个开关的输出状态(0: 未输出, 1: 输出)。	表示 n 个总加组的控制输出状态。
5	越限告警状态::=array 一个总加组告警输出状态 一个总加组告警输出状态::=structure { 总加组对象 OI, 告警输出状态 enum { 未告警(0), 告警(1) } }	

控制类方法说明见表171。

表 173 控制类方法说明

编号	方法	说明
1	复位(参数) 参数::=integer(0)	通用方法, instance-specific。
2	执行(参数) 参数::=Data	通用方法, instance-specific。
3	添加控制方案单元(控制方案单元) 控制方案单元::=instance-specific	

编号	方法	说明
4	删除控制方案单元（总加组对象） 总加组对象::=OI	
5	更新控制方案单元（控制方案单元） 控制方案单元::=instance-specific	
6	控制投入（总加组对象） 总加组对象::=OI	
7	控制解除（总加组对象） 总加组对象::=OI	

8.2.15 区间统计类

本接口类定义了统计越限相关的信息，定义见表172。

表 174 区间统计类定义

区间统计类, class_id=14		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 统计结果表	(dyn.)	array
3. 关联对象属性表	(static)	array
方法		必选/可选
1. 复位		可选
2. 执行		可选
3. 添加一个统计对象		可选
4. 删除一个统计对象		可选

区间统计类属性说明见表173。

表 175 区间统计类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	统计结果表::=array 一个统计结果 一个统计结果::=structure { 对象属性描述符 OAD, 区间统计值 array 一个统计区间 } 一个统计区间::=structure { 累计时间 double-long-unsigned, 累计次数 double-long-unsigned }	“关联对象属性表”配置的对象属性统计结果值。 累计时间：单位为秒。
3	关联对象属性表::=array 一个关联对象属性	统计间隔：采样点取值间隔。

编号	属性	说明
	一个关联对象属性 ::= structure { 关联对象属性描述符 OAD, 越限判断参数 array Data, 统计周期 unsigned, 统计间隔 TI }	

区间统计类方法说明见表174。

表 176 区间统计类方法说明

编号	方法	说明
1	复位 (参数) 参数 ::= integer (0)	通用方法, instance-specific。
2	执行 (参数) 参数 ::= Data	通用方法, instance-specific。
3	添加一个统计对象 (参数) 参数 ::= structure { 关联对象属性描述符 OAD, 越限判断参数 array Data, 统计周期 unsigned, 统计频率 TI }	向属性“关联对象属性表”中, 增加一个统计对象。
4	删除一个统计对象 (参数) 参数 ::= OAD 关联对象属性描述符	从属性“关联对象属性表”中, 删除一个统计对象。

8.2.16 累加平均类

本接口类定义了对相同属性的数值进行累加、平均的运算功能, 定义见表175。

表 177 累加平均类定义

累加平均类, class_id=15		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 运算结果	(dyn.)	structure
3. 关联对象属性表	(static)	array
方法		可选/可选
1. 复位		可选
2. 执行		可选
3. 添加一个关联对象		可选
4. 删除一个关联对象		可选

累加平均类属性说明见表176。

表 178 累加平均类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	运算结果::=array structure { 对象属性描述符 OAD, 累加和 instance-specific, 平均值 instance-specific }	累加和、平均值——其数据类型由“逻辑名”引用的对象实例定义。
3	关联对象属性表::=array 一个关联的对象 一个关联的对象::=structure { 关联对象属性描述符 OAD, 统计周期 unsigned, 统计间隔 TI }	

累加平均类方法说明见表177。

表 179 累加平均类方法说明

编号	方法	说明
1	复位（参数） 参数::=integer (0)	通用方法，instance-specific。
2	执行（参数） 参数::=Data	通用方法，instance-specific。
3	添加一个关联对象（参数） 参数::=structure { 关联对象属性描述符 OAD, 统计周期 unsigned, 统计频率 TI }	向属性“关联对象属性表”中，增加一个关联对象。
4	删除一个关联对象（参数） 参数::=OAD 关联对象属性描述符	从属性“关联对象属性表”中，删除一个关联对象。

8.2.17 极值统计类

本接口类定义了生成最大值、最小值及发生时间的功能，定义见表178。

表 180 极值统计类定义

极值工具类, class_id=16		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 极值结果表	(dyn.)	array
3. 关联对象属性表	(static)	array
方法		必选/可选
1. 复位		可选
2. 执行		可选
3. 添加一个关联对象		可选
4. 删除一个关联对象		可选

极值统计类属性说明见表179。

表 181 极值统计类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	极值结果表::=array 一个极值结果 一个极值结果::=structure { 对象属性描述符 OAD, 最大值 instance-specific, 发生时间 date_time_s, 最小值 instance-specific, 发生时间 date_time_s }	最大值、最小值——其数据类型由“逻辑名”引用的对象实例定义。
3	关联对象属性表::=array 一个关联的对 象属性 一个关联对象::=structure { 关联对象属性描述符 OAD, 统计周期 unsigned, 统计间隔 TI }	

极值统计类方法说明见表180。

表 182 极值统计类方法说明

编号	方法	说明
1	复位 (参数) 参数::=integer (0)	通用方法, instance-specific。
2	执行 (参数) 参数::=Data	通用方法, instance-specific。
3	添加一个关联对象 (参数)	向属性“关联对象属性表”中, 增加一个关联对象。

编号	方法	说明
	参数::=structure { 关联对象属性描述符 OAD, 统计周期 unsigned, 统计间隔 TI }	
4	删除一个关联对象（参数） 参数::=OAD 关联对象属性描述符	从属性“关联对象属性表”中，删除一个关联对象。

8.2.18 显示类

本接口类定义了与显示相关的信息，定义见表181。

表 183 显示类定义

显示类, class_id=17		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 显示对象列表	(static)	array
3. 显示时间	(static)	long-unsigned
4. 显示参数	(static)	structure
方法		必选/可选
1. 复位		可选
2. 执行		可选
3. 下翻		可选
4. 上翻		可选
5. 显示查看		可选
6. 全显		可选

显示类属性说明见表182。

表 184 显示类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	显示对象列表::=array 显示对象描述符 显示对象描述符::=structure { 显示对象 CSD, 屏序号 unsigned }	显示对象的数据依次被循环显示；屏序号：0表示不需要分屏；1表示分屏第一屏，2表示分屏第二屏，以此类推。
3	每个对象显示时长: long-unsigned	显示时长——单位为秒，0表示由外部触发。
4	显示参数::=structure { 当前总对象数 unsigned, }	当前总对象数：当前显示的总对象数。 允许最大对象数：可设置的最大显示数。

编号	属性	说明
	允许最大对象数 unsigned }	

显示类方法说明见表183。

表 185 显示类方法说明

编号	方法	说明
1	复位 (参数) 参数::=integer (0)	通用方法, instance-specific。
2	执行 (参数) 参数::=Data	通用方法, instance-specific。
3	下翻 (参数) 参数::=NULL	显示下一个对象信息。
4	上翻 (参数) 参数::=NULL	显示上一个对象信息。
5	显示查看 (参数) 参数::=structure { 显示列信息 CSD, 屏序号 unsigned, 显示持续时间 long-unsigned }	参数可以为所有可显示的对象。 显示持续时间——单位: 秒。
6	全显 (参数) 参数::=long-unsigned	参数为全显持续时间, 单位: 秒。

8.2.19 文件传输类

本接口类定义了服务器实现上传和下载文件的功能, 定义见表184。

表 186 文件传输类定义

文件传输类, class_id=18		0…n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 文件信息	(dyn.)	structure
3. 文件传输状态	(dyn.)	enum
方法		必选/可选
1. 复位		可选
2. 执行		可选
3. 删除		可选
4. 校验		可选
5. 代发		可选

文件传输类属性说明见表185。

表 187 文件传输类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	文件信息::=structure { 源文件 visible-string, 目标文件 visible-string, 文件大小 double-long-unsigned, 文件属性 bit-string(SIZE(3)), 文件版本 visible-string, 文件类别 enum { 当前设备文件 (0), 其他设备文件 (1) } } 文件属性: bit0: 读 (1: 可读, 0: 不可读); bit1: 写 (1: 可写, 0: 不可写); bit2: 执行 (1: 可执行, 0: 不可执行)。	源文件: 文件路径及文件名称; 目标文件: 文件路径及文件名称; 文件路径及文件名称格式: “[/路径/]文件名”。 文件大小: 单位字节。
3	文件传输状态::=enum { 文件传输进度 0...99% (0...99), 传输或执行操作成功 (100), 扩展传输正在建立连接 (101), 扩展传输正在远程登录 (102), 正在执行文件 (103), 文件或目录不存在 (104), 操作不允许 (创建/删除/读写/执行) (105), 文件传输中断 (106), 文件校验失败 (107), 文件转发失败 (108), 文件代收失败 (109), 扩展传输建立连接失败 (110), 扩展传输远程登录失败 (111), 存储空间不足 (112), 复位后默认值 (255) }	当前文件传输的过程状态。

文件传输类方法说明见表186。

表 188 文件传输类方法说明

编号	方法	说明
1	复位 (参数) 参数::=integer (0)	通用方法, instance-specific。
2	执行 (参数) 参数::=Data	通用方法, instance-specific。 执行下载文件, 如果文件没有执行权限拒绝执行。
3	删除 (参数) 参数::=null	删除本地文件, 如果文件存在则删除, 并复位文件传输状态, 否则拒绝。
4	校验 (参数) 参数::=structure { 校验文件名 visible-string, 校验类型 enum { CRC 校验 (默认) (0), md5 校验 (1), SHA1 校验 (2) }, 校验起始偏移地址 double-long-unsigned, 校验结束偏移地址 double-long-unsigned } 校验结束偏移地址为 FFFFFFFFH 表示到文件末尾。 应答::=octet-string	CRC 校验算法见附录 D。
5	代发 (参数) 参数::=CHOICE { OAD [81], ROAD [82], TSA [85], MS [92] }	文件下载到本地端后, 根据目标地址或端口进行文件代发。

8.2.20 设备管理类

本接口类定义了设备管理相关信息, 定义见表187。

表 189 设备管理类定义

设备管理类, class_id=19		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 设备描述符	(static)	visible-string
3. 版本信息	(static)	structure
4. 生产日期	(static)	date_time_s
5. 子设备列表	(static)	array
6. 支持规约列表	(static)	array

设备管理类, class_id=19		0...n
7. 允许跟随上报	(static)	bool
8. 允许主动上报	(static)	bool
9. 允许与主站通话	(static)	bool
10. 上报通道	(static)	array
11. 服务授权状态	(dyn.)	array
12. 设备运行信息	(dyn.)	array
方法		必选/可选
1. 复位		必选
2. 执行		可选
3. 数据初始化		必选
4. 恢复出厂参数		必选
5. 事件初始化		必选
6. 需量初始化		可选
7. 服务授权开启		可选

设备管理类属性说明见表188。

表 190 设备管理类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	设备描述符::=visible-string	
3	版本信息::=structure <pre> { 厂商代码 visible-string(SIZE(4)), 软件版本号 visible-string(SIZE(4)), 软件版本日期 visible-string(SIZE(6)), 硬件版本号 visible-string(SIZE(4)), 硬件版本日期 visible-string(SIZE(6)), 厂商扩展信息 visible-string(SIZE(8)) } </pre>	
4	生产日期::=date_time_s	
5	子设备列表::= array 0I	包含设备自身的子模块信息。
6	支持规约列表::=array visible-string	
7	允许跟随上报::=bool	True: 允许跟随上报, False: 禁止跟随上报。
8	允许主动上报::=bool	True: 允许主动上报, False: 禁止主动上报。
9	允许与主站通话::=bool	True: 允许通话, False: 禁止通话。
10	上报通道::= array 0AD	
11	服务器授权状态::= array 授权状态 授权状态::=structure <pre> { 服务类别 unsigned, 授权开启剩余时长 unsigned } </pre>	授权开启剩余时长单位: 分钟

编号	属性	说明
	}	
12	<pre> 设备运行信息 ::= structure { CPU 核数 unsigned, CPU 主频 long-unsigned (单位: GHz, 换算: -2), CPU 缓存 long-unsigned (单位: MB, 换算: -2), CPU 架构 visible-string, 物理内存 double-long-unsigned (单位: KB, 换算: 0), 虚拟内存 double-long-unsigned (单位: KB, 换算: 0), 磁盘总空间 double-long-unsigned (单位: KB, 换算: 0), 内存使用率 long-unsigned (单位: %, 换算: -2), 硬盘使用率 long-unsigned (单位: %, 换算: -2), CPU 使用率 long-unsigned (单位: %, 换算: -2), 操作系统名称 visible-string, 操作系统版本 visible-string, 操作系统内核 visible-string, 系统启动时间 date_time_s } </pre>	

设备管理类方法说明见表189。

表 191 设备管理类方法说明

编号	方法	说明
1	复位 (参数) 参数 ::= NULL	设备复位重启。
2	执行 (参数) 参数 ::= Data	通用方法, instance-specific。
3	数据初始化 (参数) 参数 ::= NULL	清空设备数据区。
4	恢复出厂参数 (参数) 参数 ::= array OAD	将设备的配置恢复到出厂设置。 参数定义的 OAD 列表, 保持现状, 不在恢复之列。
5	事件初始化 (参数) 参数 ::= NULL	清空所有事件存储区。
6	需量初始化 (参数) 参数 ::= NULL	当前需量对象清零。
7	服务授权开启 (参数) 参数 ::= structure { 服务类别 unsigned, 授权开启时长 unsigned }	授权开启时长单位: 分钟

8.2.21 应用连接类

本接口类定义了应用连接相关信息，定义见表190。

表 192 应用连接类定义

应用连接类, class_id=20		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 对象列表	(static)	array
3. 应用语境信息	(static)	structure
4. 当前连接的客户机地址	(dyn.)	unsigned
5. 身份验证机制	(static)	enum
方法		必选/可选
1. 复位		可选
2. 执行		可选

应用连接类属性说明见表191。

表 193 应用连接类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	对象列表::=array 一个可访问对象 一个可访问对象::=structure { 对象标识 OI, 访问权限 structure } 访问权限::=structure { 属性访问权限 array 一个属性访问权限, 方法访问权限 array 一个方法访问权限 } 一个属性访问权限::=structure { 属性 ID unsigned, 属性访问权限类别 enum { 不可访问 (0), 只读 (1), 只写 (2), 可读写 (3) } } 一个方法访问权限::=structure	包含所有可访问对象以及该对象属性和方法的访问权限。

编号	属性	说明
	{ 方法 ID unsigned, 方法访问权限 bool }	
3	应用语境信息 ::= structure { 协议版本信息 long-unsigned, 最大接收帧尺寸 long-unsigned, 最大发送帧尺寸 long-unsigned, 最大可处理帧尺寸 long-unsigned, 协议一致性块 bit-string(64), 功能一致性块 bit-string(128), 静态超时时间 double-long-unsigned }	
4	当前连接的客户机地址 ::= unsigned	
5	连接认证机制 ::= enum { 公共连接 (0), 普通密码 (1), 对称加密 (2), 数字签名 (3) }	见 7.4.3.2 。

应用连接类方法说明见表192 。

表 194 应用连接类方法说明

编号	方法	说明
1	复位 (参数) 参数 ::= integer (0)	通用方法, instance-specific。
2	执行 (参数) 参数 ::= Data	通用方法, instance-specific。

8.2.22 ESAM 接口类

本接口类定义了ESAM相关接口, 定义见表193 。

表 195 ESAM 接口类定义

ESAM 接口类, class_id=21		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. ESAM 序列号	(static)	octet-string
3. ESAM 版本号	(static)	octet-string
4. 对称密钥版本	(static)	octet-string

ESAM 接口类, class_id=21		0...n
5. 会话时效门限	(static)	double-long-unsigned
6. 会话时效剩余时间	(dyn.)	double-long-unsigned
7. 当前计数器	(static)	structure
8. 证书版本	(static)	structure
9. 终端证书序列号	(static)	octet-string
10. 终端证书	(static)	octet-string
11. 主站证书序列号	(static)	octet-string
12. 主站证书	(static)	octet-string
13. ESAM 安全存储对象列表	(static)	array
14. 红外认证时效门限	(static)	double-long-unsigned
15. 红外认证剩余时间	(dyn.)	double-long-unsigned
<i>方法</i>		<i>必选/可选</i>
1. 复位		可选
2. 执行		可选
3. ESAM 数据读取		可选
4. 数据更新		可选
5. 协商失效		可选
6. 钱包操作（开户、充值、退费）		可选
7. 密钥更新		可选
8. 证书更新		可选
9. 设置协商时效		可选
10. 钱包初始化		可选
11. 红外认证请求		可选
12. 红外认证指令		可选

ESAM接口类属性说明见表194。

表 196 ESAM 接口类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	ESAM 序列号::=octet-string	ESAM 唯一的标识，是一串数字。
3	ESAM 版本号::=octet-string	ESAM 的版本号。
4	对称密钥版本::=octet-string	ESAM 中对称密钥的版本号。
5	会话时效门限::=double-long-unsigned	单位：分钟。
6	会话时效剩余时间::=double-long-unsigned	单位：分钟。
7	当前计数器::=structure { 单地址应用协商计数器 double-long-unsigned, 主动上报计数器 double-long-unsigned, 应用广播通信序列号 double-long-unsigned }	
8	证书版本::=structure	

编号	属性	说明
	{ 终端证书版本 octet-string, 主站证书版本 octet-string }	
9	终端证书序列号::=octet-string	
10	终端证书::=octet-string	
11	主站证书序列号::=octet-string	
12	主站证书::=octet-string	主站带 MAC 下发, 同主站证书一起下发 (后 4 个字节为 MAC)。
13	ESAM 安全存储对象列表::=array OAD	需要存储到 ESAM 中的对象。
14	红外认证时效门限::=double-long-unsigned	单位: 分钟。
15	红外认证剩余时间::=double-long-unsigned	单位: 分钟。

ESAM接口类方法说明见表195。

表 197 ESAM 接口类方法说明

编号	方法	说明
1	复位 (参数) 参数::=integer (0)	通用方法, instance-specific。
2	执行 (参数) 参数::=Data	通用方法, instance-specific。
3	ESAM 数据读取 (参数) 参数::=SID 应答::=octet-string	
4	数据更新 (参数) 参数::=structure { 参数内容 octet-string, 数据验证码 SID_MAC } 参数内容格式: 4 字节 OAD+ 1 字节参数值长度 + 参数值。	数据、数据 MAC 先发给 ESAM 验证, 验证成功后, 再设置到终端中。
5	协商失效 (参数) 参数::=NULL 应答::= date_time_s	应答时间为当前日期时间。
6	钱包操作 (参数) 参数::=structure { 操作类型 integer, 购电金额 double-long-unsigned, 购电次数 double-long-unsigned,	开户时, 不需要验证客户编号, 直接将客户编号写到 ESAM 的对应文件中; 再进行充值操作。

编号	方法	说明
	户号 octet-string, 数据验证码 SID_MAC, 表号 octet-string } 操作类型: 0 开户; 1 充值; 2 退费。	
7	密钥更新 (参数) 参数::=structure { 密钥密文 octet-string, 数据验证码 SID_MAC }	
8	证书更新 (参数) 参数::=structure { 证书内容 octet-string, 安全标识 SID }	
9	设置协商时效 (参数) 参数::=structure { 参数内容 octet-string, 安全标识 SID }	
10	钱包初始化 (参数) 参数::=structure { 预置金额 double-long-unsigned, 数据验证码 SID_MAC }	
11	红外认证请求 (参数) 参数::=RN (随机数 1) 应答::=structure { 表号 octet-string, ESAM 序列号 octet-string, 随机数 1 密文 octet-string, 随机数 2 RN }	
12	红外认证指令 (参数) 参数::=随机数 2 密文 octet-string	

8.2.23 输入输出设备类

本接口类定义了输入输出设备相关信息，定义见表196。

表 198 输入输出设备类定义

输入输出设备类, class_id=22		0…n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 设备对象列表	(static)	array
3. 设备对象数量	(static)	unsigned
4. 配置参数	(static)	instance-specific
5. 端口授权状态	(dyn.)	array
方法		必选/可选
1. 复位		可选
2. 执行		可选
3. 端口授权开启		可选

输入输出设备类的属性描述见表197。

表 199 输入输出设备类属性描述

编号	属性	说明
1	逻辑名 ::= octet-string	
2	设备对象列表 ::= array instance-specific	设备对象数据类型由“逻辑名”引用的对象实例定义。
3	设备对象数量 ::= unsigned	当前设备对象数量。
4	配置参数 ::= instance-specific	数据格式由“逻辑名”引用的对象实例定义。
5	端口授权状态 ::= array 授权开启剩余时长 授权开启剩余时长 ::= unsigned	授权开启剩余时长单位：分钟

输入输出设备类的方法描述见表198。

表 200 输入输出设备类方法描述

编号	方法	说明
1	复位 (参数) 参数 ::= bit-string	通用方法, instance-specific。
2	执行 (参数) 参数 ::= Data	通用方法, instance-specific。
3	端口授权开启 (参数) 参数 ::= structure { 端口序号 unsigned, 授权开启时长 unsigned }	授权开启时长单位：分钟。

8.2.24 总加组类

本接口类定义了总加组相关信息，定义见表199。

表 201 总加组类定义

总加组类, class_id=23		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 总加组配置表	(static)	array
3. 总加组有功功率	(dyn.)	long64
4. 总加组无功功率	(dyn.)	long64
5. 总加组滑差时间内平均有功功率	(dyn.)	long64
6. 总加组滑差时间内平均无功功率	(dyn.)	long64
7. 总加组日有功电量	(dyn.)	array
8. 总加组日无功电量	(dyn.)	array
9. 总加组月有功电量	(dyn.)	array
10. 总加组月无功电量	(dyn.)	array
11. 总加组剩余电量(费)	(dyn.)	long64
12. 当前功率下浮控制后总加组有功功率冻结值	(dyn.)	long64
13. 总加组滑差时间周期	(static)	unsigned
14. 总加组功控轮次配置	(static)	bit-string
15. 总加组电控轮次配置	(static)	bit-string
16. 总加组控制设置状态	(dyn.)	structure
17. 总加组当前控制状态	(dyn.)	structure
18. 换算及单位	(static)	structure
方法		必选/可选
1. 清空总加组配置单元		必选
2. 执行		可选
3. 添加一个总加组配置单元		可选
4. 批量添加总加组配置单元		可选
5. 删除一个总加组配置单元		可选

总加组类的属性描述见表200。

表 202 总加组类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	总加组配置表::=array 总加组配置单元 总加组配置单元::=structure { 参与总加的分路通信地址 TSA, 总加标志 enum{正向(0), 反向(1)}, 运算符标志 enum{加(0), 减(1)} }	
3	总加组有功功率::=long64	

编号	属性	说明
4	总加组无功功率::=long64	
5	总加组滑差时间内平均有功功率::=long64	
6	总加组滑差时间内平均无功功率::=long64	
7	总加组日有功电量::=array 电能量 电能量::=long64	总及 n 个费率的电能量，第一个数组元素是总电能量，从第二个数组元素依次排列费率 1...n 电能量。
8	总加组日无功电量::=array 电能量 电能量::=long64	同上。
9	总加组月有功电量::=array 电能量 电能量::=long64	同上。
10	总加组月无功电量::=array 电能量 电能量::=long64	同上。
11	总加组剩余电量(费)::=long64	
12	当前功率下浮控控后总加组有功功率冻结值::=long64	
13	总加组滑差时间周期::=unsigned	单位：分。
14	总加组功控轮次配置::=bit-string (SIZE(8)) bit0...bit7 按顺序对位表示第 1...第 8 轮次开关的受控设置，置“1”表示该轮次开关受控，置“0”表示不受控。	
15	总加组电控轮次配置::=bit-string (SIZE(8)) bit0...bit7 按顺序对位表示第 1...第 8 轮次开关的受控设置，置“1”表示该轮次开关受控，置“0”表示不受控。	
16	总加组控制设置状态::=structure { 时段控定值方案号 unsigned, 功控时段有效标志位 bit-string(SIZE(8)), 功控状态 PCState, 电控状态 ECState, 功控轮次状态 TrunState, 电控轮次状态 TrunState } 功控时段有效标志位：bit0...bit7 按顺序对位表示 1...8 时段控投入的有效时段，置“1”：有效，置“0”：无效。 PCState::=bit-string(SIZE(8)) bit0...bit7 按顺序对位表示： 置“1”：投入，置“0”：解除； bit0：时段控； bit1：厂休控； bit2：营业报停控； bit3：当前功率下浮控； bit4...bit7：备用。 ECState::=bit-string(SIZE(8))	时段控定值方案号：表示所投入的功控定值方案号。

编号	属性	说明
	bit0...bit7 按顺序对位表示； 置“1”：投入，置“0”：解除； bit0：月电控； bit1：购电控； bit2...bit7：备用。 TrunState ::= bit-string(SIZE(8)) bit0...bit7 按顺序对位表示 1...8 轮次开关的受控状态；置“1”： 受控，置“0”：不受控。	
17	总加组当前控制状态 ::= structure { 当前功控定值 long64 (单位: W 换算: -1), 当前功率下浮控浮动系数 integer (单位: %), 功控跳闸输出状态 OutputState, 月电控跳闸输出状态 OutputState, 购电控跳闸输出状态 OutputState, 功控越限告警状态 PCAlarmState, 电控越限告警状态 ECArmState } OutputState ::= bit-string(SIZE(8)) bit0...bit7 分别表示终端 1...8 轮次跳闸输出状态，置“1”：处 于跳闸状态，置“0”：未处于跳闸状态。 PCAlarmState ::= bit-string(SIZE(8)) 按顺序对位表示；置“1”：处于某种功控越限告警状态； 置“0”：未处于告警状态； bit0：时段控； bit1：厂休控； bit2：营业报停控； bit3：当前功率下浮控； bit4...bit7：备用。 ECArmState ::= bit-string(SIZE(8)) 置“1”：处于某种电控越限告警状态； 置“0”：未处于告警状态； bit0：月电控； bit1：购电控； bit2...bit7：备用。	
18	换算及单位 ::= structure { 属性 3 单位换算 Scaler_Unit (单位: W, 换算: -1), 属性 4 单位换算 Scaler_Unit (单位: var, 换算: -1), 属性 5 单位换算 Scaler_Unit (单位: W, 换算: -1), 属性 6 单位换算 Scaler_Unit (单位: var, 换算: -1), }	Scaler_Unit 见表 17。

编号	属性	说明
	属性 7 单位换算 Scaler_Unit (单位: kWh, 换算: -4), 属性 8 单位换算 Scaler_Unit (单位: kvarh, 换算: -4), 属性 9 单位换算 Scaler_Unit (单位: kWh, 换算: -4), 属性 10 单位换算 Scaler_Unit (单位: kvarh, 换算: -4), 属性 11 单位换算 Scaler_Unit (单位: kWh/元, 换算: -4), 属性 12 单位换算 Scaler_Unit (单位: W, 换算: -1) }	

总加组类的方法描述见表 201 。

表 203 总加组类方法说明

编号	方法	说明
1	清空总加配置表(参数) 参数::=NULL	清空总加组配置表。
2	执行(参数) 参数::=Data	通用方法, instance-specific。
3	添加一个总加组配置单元(参数) 参数::=总加组配置单元	向总加组配置表中添加一个总加配置单元。
4	批量添加总加组配置单元(参数) 参数::=array 总加组配置单元	向总加组配置表中添加若干个总加配置单元。
5	删除一个总加组配置单元(参数) 参数::=参与总加的分路通信地址 TSA	删除总加组配置表中的一个总加组配置单元。

8.2.25 分项事件对象类

本接口类定义了配置、存储分项事件数据信息, 定义见表202 。

表 204 分项事件对象类定义

分项事件对象类, class_id=24		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 关联对象属性表	(static)	array
3. 当前记录数	(dyn.)	structure
4. 最大记录数	(static)	long-unsigned
5. 配置参数	(static)	structure
6. 事件记录表 1	(dyn.)	array
7. 事件记录表 2	(dyn.)	array
8. 事件记录表 3	(dyn.)	array
9. 事件记录表 4	(dyn.)	array
10. 当前值记录表	(dyn.)	array
11. 上报标识	(static)	enum
12. 有效标识	(static)	bool
14. 时间状态记录表	(dyn.)	array
方法		必选/可选

分项事件对象类, class_id=24	0...n
1. 复位	可选
2. 执行	可选
4. 添加一个事件关联对象属性	可选
5. 删除一个事件关联对象属性	可选

分项事件对象类属性说明见表203。

表 205 分项事件对象类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	关联对象属性表::=array 一个关联的对象属性 一个关联的对象属性::=OAD	事件记录中的关联对象属性数据依据关联对象属性表产生。 关联对象属性的属性标识见表 152。
3	当前记录数::=structure { 记录表 1 当前记录数 long-unsigned, 记录表 2 当前记录数 long-unsigned, 记录表 3 当前记录数 long-unsigned, 记录表 4 当前记录数 long-unsigned, }	表示保存在事件记录表中的记录数。执行“复位”方法后,记录表清空,此时当前记录数为 0。
4	最大记录数::=long-unsigned	事件记录表最小存储条数。
5	配置参数::=structure { 参数 1 instance-specific, ... 参数 n instance-specific }	配置触发事件记录的判定参数,参数的数据类型由“逻辑名”引用的对象实例定义。见 7.3.5。
6	事件记录表 1::=array 分项事件记录单元 分项事件记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件上报状态 array 通道上报状态, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }	用于存储事件记录,记录根据“配置参数”自动产生。 记录的顺序按照事件发生的次序排序。执行“复位”方法后事件记录表清空。 事件记录序号——单调递增。 事件上报状态——见表 152。 关联对象属性的数据——其排列次序和数量 n,由“逻辑名”引用的对象实例的属性 2 定义。
7	事件记录表 2::=array 分项事件记录单元	
8	事件记录表 3::=array 分项事件记录单元	
9	事件记录表 4::=array 分项事件记录单元	
10	当前值记录表::=array structure	包含事件发生的次数以及累计时间。

编号	属性	说明
	{ 事件发生次数 double-long-unsigned, 事件累计时间 double-long-unsigned, }	事件发生次数——单位：次 事件累计时间——单位：秒 数组元素按顺序对应：事件记录表 1、事件记录表 2、事件记录表 3、事件记录表 4，如无该记录表则该元素为 NULL。
11	上报标识::=enum { 不上报 (0), 事件发生上报 (1), 事件恢复上报 (2), 事件发生恢复均上报 (3) }	
12	有效标识::=bool	
14	时间状态记录表::=array structure { 最近一次发生时间 date_time_s, 最近一次结束时间 date_time_s }	最近一次发生时间——取上 1 次事件的发生时间，如果未发生返回 NULL。 最近一次结束时间——上 1 次事件未结束，则取上 2 次事件的结束时间；不存在上 2 次事件或结束时间不支持，则返回 NULL。

分项事件对象类方法说明见表204。

表 206 分项事件对象类方法说明

编号	方法	说明
1	复位 (参数) 参数::=integer (0)	通用方法, instance-specific。
2	执行 (参数)	通用方法, instance-specific。
3	参数::=Data	
4	添加一个事件关联对象属性 (参数) 参数::=OAD 对象属性描述符	向属性“关联对象属性表”中，增加一个关联对象属性。
5	删除一个事件关联对象属性 (参数) 参数::=OAD 对象属性描述符	从属性“关联对象属性表”中，删除一个关联对象属性。

8.2.26 无线公网/专网通信接口类

本接口类定义了无线公网/专网通信接口，定义见表205。

表 207 无线公网/专网通信接口类定义

无线公网通信接口类, class_id=25		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 通信配置	(static)	structure
3. 主站通信参数表	(static)	array
4. 短信通信参数	(static)	structure

无线公网通信接口类, class_id=25		0...n
5. 版本信息	(static)	structure
6. 支持规约列表	(static)	array
7. SIM 卡 ICCID	(static)	visible-string(SIZE(20))
8. IMSI	(static)	visible-string(SIZE(15))
9. 信号强度	(dyn)	long, 单位: dBm
10. SIM 卡号码	(dyn.)	visible-string(SIZE(16))
11. 终端 IP	(dyn)	octet-string
12. 设备描述符	(dyn)	visible-string
13. 运营商及网络制式	(dyn)	structure
14. 多网络配置	(static)	array
方法		必选/可选
1. 复位		可选

无线公网/专网通信接口类属性说明见表206。

表 208 无线公网/专网通信接口类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	通信配置::=structure { 工作模式 enum{混合模式(0), 客户机模式(1), 服务器模式(2)}, 在线方式 enum{永久在线(0), 被动激活(1)}, 连接方式 enum{TCP(0), UDP(1)}, 连接应用方式 enum{主备模式(0), 多连接模式(1)}, 侦听端口列表 array long-unsigned, APN visible-string, 用户名 visible-string, 密码 visible-string, 代理服务器地址 octet-string, 代理端口 long-unsigned, 超时时间及重发次数 unsigned (bit0...bit1: 重发次数, bit2...bit7: 超时时间(秒)), 心跳周期(秒) long-unsigned }	
3	主站通信参数表::=array 主站通信参数 主站通信参数::=structure { IP 地址 octet-string, 端口 long-unsigned }	

编号	属性	说明
4	短信通信参数 ::= structure { 短信中心号码 visible-string(SIZE(16)), 主站号码 array visible-string(SIZE(16)), 短信通知目的号码 array visible-string(SIZE(16)) }	
5	版本信息 ::= structure { 厂商代码 visible-string(SIZE(4)), 软件版本号 visible-string(SIZE(4)), 软件版本日期 visible-string(SIZE(6)), 硬件版本号 visible-string(SIZE(4)), 硬件版本日期 visible-string(SIZE(6)), 厂商扩展信息 visible-string(SIZE(8)) }	
6	支持规约列表 ::= array visible-string	
7	SIM 卡 ICCID ::= visible-string(SIZE(20))	
8	IMSI ::= visible-string(SIZE(15))	
9	信号强度 ::= long	单位: dBm。
10	SIM 卡号码 ::= visible-string(SIZE(16))	
11	终端 IP ::= octet-string	
12	设备描述符 ::= visible-string	
13	运营商及网络制式 ::= structure { 运营商 unsigned, 网络制式 unsigned }	
14	多网络配置 ::= array structure { 运营商代码 enum{CMCC(0), CTCC(1), CUCC(2), 未知(255)}, 网络类型代码 enum{2G(0), 3G(1), 4G(2), 5G(3), 未知(255)}, APN visible-string, 用户名 visible-string, 密码 visible-string, 代理服务器地址 octet-string, 代理端口 long-unsigned, 鉴权方式 enum {null(0), chap(1), pap(2), pap&chap(3), 自适应(255)}, 主站通信参数 array structure { IP 地址 octet-string, }	

编号	属性	说明
	端口 long-unsigned	

无线公网通信接口类方法说明见表207。

表 209 无线公网通信接口类方法说明

编号	方法	说明
1	复位 (参数) 参数 ::= NULL	复位无线公网通信接口。

8.2.27 以太网通信接口类

本接口类定义了以太网通信接口，定义见表208。

表 210 以太网通信接口类定义

以太网通信接口类, class_id=26		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 通信配置	(static)	structure
3. 主站通信参数表	(static)	array
4. 终端 IP	(static)	structure
5. MAC 地址	(static)	octet-string
方法		必选/可选
1. 复位		可选

以太网通信接口类属性说明见表209。

表 211 以太网通信接口类属性说明

编号	属性	说明
1	逻辑名 ::= octet-string	
2	通信配置 ::= structure { 工作模式 enum{混合模式(0), 客户机模式(1), 服务器模式(2)}, 连接方式 enum{TCP(0), UDP(1)}, 连接应用方式 enum{主备模式(0), 多连接模式(1)} 侦听端口列表 array long-unsigned, 代理服务器地址 octet-string, 代理端口 long-unsigned, 超时时间及重发次数 unsigned (bit0...bit1: 重发次数,)	连接应用方式: 主备模式: 按主站通信参数次序依次切换连接主站, 只连接一个主站; 多连接模式: 同时连接多个主站。

编号	属性	说明
	bit2…bit7: 超时时间 (秒)), 心跳周期(秒) long-unsigned }	
3	主站通信参数表::=array 主站通信参数 主站通信参数::=structure { IP 地址 octet-string, 端口 long-unsigned }	
4	网络配置::=structure { IP 配置方式 enum{DHCP (0), 静态 (1), PPPoE (2) }, IP 地址 octet-string, 子网掩码 octet-string, 网关地址 octet-string, PPPoE 用户名 visible-string, PPPoE 密码 visible-string }	
5	MAC 地址::=octet-string	

以太网通信接口类方法说明见表210。

表 212 以太网通信接口类方法说明

编号	方法	说明
1	复位 (参数) 参数::=NULL	复位以太网接口。

8.2.28 监控单元集合接口类

本接口类定义了监控单元集合接口，定义见表211。

表 213 监控单元集合接口类定义

监控单元集合接口类, class_id=27		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 监控单元配置表	(static)	array
3. 当前遥信量集合	(dyn.)	array
4. 当前遥测量集合	(dyn.)	array
5. 参数对象集合	(static)	array
方法		必选/可选
1. 复位		可选
2. 执行		可选

监控单元集合接口类, class_id=27	0...n
3. 添加一个监控单元	必选
4. 批量添加监控单元	必选
5. 删除一个监控单元	必选
6. 参数配置	可选
7. 遥控选择	必选
8. 遥控执行	必选
9. 遥控取消	必选

监控单元集合接口类属性说明见表212。

表 214 监控单元集合接口类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	监控单元配置表::=array 监控单元 监控单元::=structure	
2	{ 监控单元地址 TSA, 遥信对象集合 array OAD, 遥测对象集合 array OAD, 遥控对象集合 array OAD }	
3	当前遥信量集合::=array 遥信量单元 遥信量单元::=structure { 监控单元地址 TSA, 遥信对象 OAD, 遥信值 Data }	
4	当前遥测量集合::=array 遥测量单元 遥测量单元::=structure { 监控单元地址 TSA, 遥测对象 OAD, 遥测值 Data }	
5	参数对象集合::=array 参数单元 参数单元::=structure { 监控单元地址 TSA, 参数对象 OAD, 参数值 Data }	

监控单元集合接口类方法说明见表213。

表 215 监控单元集合接口类方法说明

编号	方法	说明
1	复位（参数） 参数::=NULL	清空监控单元配置表。
2	执行（参数） 参数::=Data	通用方法，instance-specific。
3	添加一个监控单元（参数） 参数::=监控单元	在监控单元配置表中添加一个监控单元。
4	批量添加监控单元（参数） 参数::=array 监控单元	在监控单元配置表中添加若干个监控单元。
5	删除一个监控单元（参数） 参数::=TSA（监控单元地址）	删除监控单元配置表中的一个监控单元。
6	参数配置（参数） 参数::=structure { 监控单元地址 TSA, 参数对象列表 array structure { 参数对象 OAD, 参数值 Data } }	对监控单元配置一个或多个参数。
7	遥控选择（参数） 参数::=structure { 监控单元地址 TSA, 遥控对象 OAD, 遥控动作类型 enum{分（1），合（2）}, }	对监控单元的遥控对象进行遥控选择命令，在进行遥控时先进行遥控选择命令，该命令必须带时间附加信息域
8	遥控执行（参数） 参数::=structure { 监控单元地址 TSA, 遥控对象 OAD, 遥控动作类型 enum{分（1），合（2）}, }	对监控单元的遥控对象进行遥控执行命令，在执行该命令前先执行遥控选择命令，该命令必须带时间附加信息域
9	遥控取消（参数） 参数::=structure { 监控单元地址 TSA, 遥控对象 OAD, 遥控动作类型 enum{分（1），合（2）}, }	对监控单元的遥控对象进行遥控撤销命令，在执行该命令前先执行遥控选择命令，该命令必须带时间附加信息域

8.2.29 电器设备接口类

本接口类定义了电器设备相关信息，定义见表214。

表 216 电器设备接口类定义

电器设备类, class_id=28		0..n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 设备描述符	(static)	visible-string
3. 厂商版本信息	(static)	structure
4. 开关机状态	(dyn.)	enum
方法		必选/可选
1. 复位		可选
2. 执行		可选
3. 开机		必选
4. 关机		必选
4. 透明转发		

电器设备接口类属性说明见表215。

表 217 电器设备接口类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	设备描述符::=visible-string	
3	厂商版本信息::=structure { 设备类型 long-unsigned, 协议版本号 visible-string, 硬件版本号 visible-string, 软件版本号 visible-string, 通信方式 bit-string, 生产厂家信息 visible-string, 设备型号 visible-string, 家电唯一标志 ID visible-string }	
4	开关机状态::=enum { 关机 (0) 开机 (1) }	

电器设备接口类方法说明见表216。

表 218 电器设备接口类方法说明

编号	方法	说明
1	复位（参数） 参数:: <code>NULL</code>	清空监控单元配置表。
2	执行（参数） 参数:: <code>Data</code>	通用方法，instance-specific。
3	开机（参数） 参数:: <code>NULL</code>	
4	关机（参数） 参数:: <code>NULL</code>	
5	透明转发（参数） 参数:: <code>octet-string</code> 返回结果:: <code>octet-string</code>	

8.2.30 自描述接口类

本接口类定义了一种自描述结构信息，定义见表211。

表 219 自描述接口类定义

自描述类，class_id=29		0...n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 属性描述符	(dyn.)	array
3. 方法描述符	(dyn.)	array
方法		必选/可选
1. 复位		可选
2. 执行		可选

自描述接口类属性说明见表212。

表 220 自描述接口类属性说明

编号	属性	说明
1	逻辑名:: <code>octet-string</code>	
2	属性描述符:: <code>array visible-string</code>	
3	方法描述符:: <code>array visible-string</code>	

自描述接口类方法说明见表213。

表 221 自描述接口类方法说明

编号	方法	说明
1	复位（参数） 参数:: <code>NULL</code>	通用方法，instance-specific。
2	执行（参数） 参数:: <code>Data</code>	通用方法，instance-specific。

8.3 对象标识

8.3.1 对象标识格式定义

对象标识（OI）由两字节组成，采用十六进制编码表示，其格式定义见图15。

对象标识（OI）															
OIA								OIB							
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
OIA1				OIA2				OIB1				OIB2			

图 15 对象标识格式定义

对象标识按字节划分为OIA和OIB，OIA按位划分为OIA1和OIA2，OIB按位划分为OIB1和OIB2。本标准采用分类编码的方式定义各个对象标识编码，未定义的对象标识编码保留。OIA1与接口类对象的对应关系见A.1。

8.3.2 OIA1=0H

OIA1=0H的对象标识定义见表220。

表 222 OIA1=0H 对象标识定义

对象标识（OI）			
OIA1	OIA2	OIB1	OIB2
0H: 电能量	0H: 总 1H: 基波 2H: 谐波	0H: 组合有功 1H: 正向有功 2H: 反向有功 3H: 组合无功 1 4H: 组合无功 2 5H: 第一象限 6H: 第二象限 7H: 第三象限 8H: 第四象限 9H: 正向视在	0H: 合相 1H: A相 2H: B相 3H: C相
	3H: 铜损 4H: 铁损 5H: 关联	0H: 总有功	

8.3.3 OIA1=1H

OIA1=1H的对象标识定义见表221。

表 223 OIA1=1H 对象标识定义

对象标识（OI）			
OIA1	OIA2	OIB1	OIB2
1H: 需量	0H: 当前 1H: 冻结周期内	0H: 组合有功 1H: 正向有功	0H: 合相 1H: A相

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
		2H: 反向有功 3H: 组合无功 1 4H: 组合无功 2 5H: 第一象限 6H: 第二象限 7H: 第三象限 8H: 第四象限 9H: 正向视在 AH: 反向视在	2H: B 相 3H: C 相

8.3.4 OIA1=2H

OIA1=2H的对象标识定义见表222。

表 224 OIA1=2H 对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
2H: 变量	0H: 计量	00H: 电压 01H: 电流 02H: 电压相角 03H: 电压电流相角 04H: 有功功率 05H: 无功功率 06H: 视在功率 07H: 一分钟平均有功功率 08H: 一分钟平均无功功率 09H: 一分钟平均视在功率 0AH: 功率因数 0BH: 电压波形失真度 0CH: 电流波形失真度 0DH: 电压谐波含有率 (总及 2...n 次) 0EH: 电流谐波含有率 (总及 2...n 次) 0FH: 电网频率 10H: 表内温度 11H: 时钟电池电压 12H: 停电抄表电池电压 13H: 时钟电池工作时间 14H: 电能表运行状态字 15H: 电能表跟随上报状态字 17H: 当前有功需量 18H: 当前无功需量	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
2H: 变量	0H: 计量	19H: 当前视在需量 1AH: 当前电价 1BH: 当前费率电价 1CH: 当前阶梯电价 1EH: 事件发生时间 20H: 事件结束时间 21H: 数据冻结时间 22H: 事件记录序号 23H: 冻结记录序号 24H: 事件发生源 25H: 事件当前值 26H: 电压不平衡率 27H: 电流不平衡率 28H: 负载率 29H: 安时值 2AH: 目标服务器地址 2CH: (当前) 钱包文件 2DH: (当前) 透支金额 2EH: 累计购电金额 31H: 月度用电量 32H: 阶梯结算用电量 40H: 控制命令执行状态字 41H: 控制命令错误状态字 42H: 外置蓝牙负荷开关状态 43H: 端子座温度分钟变化量 44H: 自监测误差 46H: 谐波电压 47H: 谐波电流 48H: 谐波有功功率 50H: 电流回路状态 5EH: 停电期间数据	
	1H: 统计	00H: 分钟区间统计 01H: 小时区间统计 02H: 日区间统计 03H: 月区间统计 04H: 年区间统计 10H: 分钟平均 11H: 小时平均 12H: 日平均 13H: 月平均	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
2H: 变量	1H: 统计	14H: 年平均 20H: 分钟极值 21H: 小时极值 22H: 日极值 23H: 月极值 24H: 年极值 31H: A相电压合格率 32H: B相电压合格率 33H: C相电压合格率 40H: 日最大有功功率及发生时间 41H: 月最大有功功率及发生时间	
	2H: 采集	00H: 通信流量 03H: 供电时间 04H: 复位次数	
	3H: 总加组	01H: 总加组 1 02H: 总加组 2 03H: 总加组 3 04H: 总加组 4 05H: 总加组 5 06H: 总加组 6 07H: 总加组 7 08H: 总加组 8	
	4H: 脉冲计量	01H: 脉冲计量点 1 02H: 脉冲计量点 2 03H: 脉冲计量点 3 04H: 脉冲计量点 4 05H: 脉冲计量点 5 06H: 脉冲计量点 6 07H: 脉冲计量点 7 08H: 脉冲计量点 8	
	5H: 水气热	00H: 累计水(热)流量 01H: 累计气流量 02H: 累计热量 03H: 热功率 04H: 累计工作时间 05H: 水温 06H: (仪表)状态 ST	
	6H: TU设备	00H: 监控单元告警状态字 01H: 监控单元运行状态字	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
2H: 变量	7H: 充电桩	00H: 充电输出电压 01H: 充电输出电流 02H: 充电导引电压 03H: 输出 PWM 占空比 04H: 充电接口温度探头温度 05H: SOC 06H: 电池 07H: 充电机环境温度 08H: BMS 09H: 当前充电状态	

8.3.5 OIA1=3H

OIA1=3H的对象标识定义见表223。

表 225 OIA1=3H 对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
3H: 事件	0H: 电能表	00H: 失压 01H: 欠压 02H: 过压 03H: 断相 04H: 失流 05H: 过流 06H: 断流 07H: 功率反向 08H: 过载 09H: 正向有功需量超限 0AH: 反向有功需量超限 0BH: 无功需量超限 0CH: 功率因数超下限 0DH: 全失压 0EH: 辅助电源掉电 0FH: 电压逆相序 10H: 电流逆相序 11H: 掉电 12H: 编程 13H: 清零 14H: 需量清零 15H: 事件清零 16H: 校时	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
3H: 事件	0H: 电能表	17H: 时段表编程 18H: 时区表编程 19H: 周休日编程 1AH: 结算日编程 1BH: 开盖 1CH: 开端钮盒 1DH: 电压不平衡 1EH: 电流不平衡 1FH: 跳闸 20H: 合闸 21H: 节假日编程 22H: 有功组合方式编程 23H: 无功组合方式编程 24H: 费率参数表编程 25H: 阶梯表编程 26H: 密钥更新 27H: 异常插卡 28H: 购电记录 29H: 退费记录 2AH: 恒定磁场干扰记录 2BH: 负荷开关误动作 2CH: 电源异常 2DH: 电流严重不平衡 2EH: 时钟故障 2FH: 计量芯片故障 30H: 通信模块变更事件 31H: 管理芯插拔事件 32H: 端子座过热报警事件 33H: 端子座温度剧变事件 34H: 端子座温度不平衡事件 35H: 误差自检测事件 36H: 升级事件 37H: 停电事件 38H: 上电事件 39H: 电压谐波总畸变率超限 3AH: 电流谐波总畸变率超限 3BH: 电能表功率因数超下限事件 3CH: 电能表广播校时事件	
	1H: 采集	00H: 终端初始化 01H: 终端版本变更	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
3H: 事件	1H: 采集	04H: 状态量变位 05H: 电能表时钟超差 06H: 终端停/上电 07H: 直流模拟量越上限 08H: 直流模拟量越下限 09H: 消息认证错误 0AH: 设备故障记录 0BH: 电能表示度下降 0CH: 电能量超差 0DH: 电能表飞走 0EH: 电能表停走 0FH: 抄表失败 10H: 月通信流量超限 11H: 发现未知电能表 12H: 跨台区电能表事件 14H: 终端对时事件 15H: 遥控跳闸记录 16H: 有功总电能量差动超限事件记录 17H: 输出回路接入状态变位记录 18H: 终端编程记录 19H: 终端电流回路异常事件 1AH: 电能表在网状态切换事件 1BH: 终端对电能表校时记录 1CH: 电能表数据变更监控记录 1DH: 电能表在线状态变化事件 1EH: 拒绝从节点入网事件记录 1FH: 参数变更事件 20H: 电流互感器异常事件 21H: TA 专用模块工况事件 40H: 安全事件变更记录 41H: STA 入网事件	
	2H: 总加组	00H: 功控跳闸记录 01H: 电控跳闸记录 02H: 购电参数设置记录 03H: 电控告警事件记录	
	3H: 通用	00H: 事件上报状态 01H: 标准事件记录单元 02H: 编程记录事件单元 03H: 发现未知电能表事件单元 04H: 跨台区电能表事件单元	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
3H: 事件	3H: 通用	05H: 功控跳闸记录单元 06H: 电控跳闸记录单元 07H: 电报告警事件单元 08H: 电能表需量超限事件单元 09H: 停/上电事件单元 0AH: 遥控事件记录单元 0BH: 有功总电能差动越限事件记录单元 0CH: 事件清零事件记录单元 0DH: 终端对电能表校时记录单元 0EH: 电能表在网状态切换事件单元 0FH: 电能表数据变更监控记录单元 10H: 电能表异常插卡记录单元 11H: 退费事件记录单元 12H: 通信模块变更事件单元 13H: 电能表时钟超差记录单元对象 14H: 电能表时段表编程事件记录单元 15H: 电能表节假日编程事件记录单元 16H: 安全事件变更记录单元 17H: 电能表在线状态变化事件单元 18H: 电流互感器异常事件单元 19H: 监控单元遥信变位事件记录单元 1AH: 监控单元遥控记录事件单元 1BH: 拒绝从节点入网事件记录单元 1CH: 参数变更事件记录单元 20H: 新增上报事件列表 21H: 充电计划生成失败记录单元 22H: 充电计划上报记录单元 23H: 充电申请事件记录单元 24H: 充电执行事件记录单元 25H: 有序充电类事件记录单元 26H: 充电订单结算记录单元 27H: 系统监测事件单元 30H: STA 入网事件单元 31H: 锁具异常事件记录单元	
	4H: 有序充电	01H: 充电计划生成失败事件 02H: 充电计划上报事件 03H: 充电申请事件 04H: 充电执行事件 05H: 充电桩异常事件 06H: 充电订单结算事件	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
3H: 事件	4H: 有序充电	07H: 充电计划下发失败事件	
	5H: TU 设备	00H: 监控单元遥信变位事件 01H: 监控单元遥控记录事件	
	6H: 物联管理	00H: 系统监测事件 01H: 容器监测事件 02H: 应用监测事件	
	7H: 智能锁具	01H: 锁具异常事件	

8.3.6 OIA1=4H

OIA1=4H的对象标识定义见表224。

表 226 OIA1=4H 对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
4H: 参变量	0H: 通用	00H: 日期时间 01H: 通信地址 02H: 表号 03H: 客户编号 04H: 设备地理位置 05H: 组地址 06H: 时钟源 07H: LCD 参数 08H: 备用套时区表切换时间 09H: 备用套日时段切换时间 0AH: 备用套分时费率切换时间 0BH: 备用套阶梯电价切换时间 0CH: 时区时段数 0DH: 阶梯数 0EH: 谐波分析次数 0FH: 密钥总条数 10H: 计量元件数 11H: 公共假日 12H: 周休日特征字 13H: 周休日采用的日时段表号 14H: 当前套时区表 15H: 备用套时区表 16H: 当前套日时段表 17H: 备用套日时段表 18H: 当前套费率电价 19H: 备用套费率电价	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
4H: 参变量	0H: 通用	1AH: 当前套阶梯电价 1BH: 备用套阶梯电价 1CH: 电流互感器变比 1DH: 电压互感器变比 1EH: 报警金额限值 1FH: 其它金额限值 20H: 报警电量限值 21H: 其它电量限值 22H: 插卡状态字 24H: 剔除 25H: 采集器远程升级结果表 26H: 采集器升级结果 26H: 采集器升级结果 27H: 电能表升级结果 30H: 电压合格率参数 31H: 逻辑设备列表 40H: TA 专用模块 41H: 电流回路监测使能 47H: 当前计划时段信息	
	1H: 计量	00H: 最大需量周期 01H: 滑差时间 02H: 校表脉冲宽度 03H: 资产管理码 04H: 额定电压 05H: 额定电流/基本电流 06H: 最大电流 07H: 有功准确度等级 08H: 无功准确度等级 09H: 电能表有功常数 0AH: 电能表无功常数 0BH: 电能表型号 0CH: ABC 各相电导系数 0DH: ABC 各相电抗系数 0EH: ABC 各相电阻系数 0FH: ABC 各相电纳系数 11H: 软件备案号 12H: 有功组合方式特征字 13H: 无功组合方式 1 特征字 14H: 无功组合方式 2 特征字 16H: 结算日	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
4H: 参变量	1H: 计量	17H: 期间需量冻结周期	
	2H: 采集	04H: 终端广播校时时间	
	3H: 设备	00H: 电气设备 07H: 水表 08H: 气表 09H: 热表 10H: 智能锁具 11H: 热水器 12H: 洗衣机 13H: 智能插座 14H: 空调 15H: 锁具状态监测器	
	4H: 应用连接	00H: 应用连接 01H: 认证密码	
	5H: 远程通信模块	00H: 公网远程通信模块 1 01H: 公网远程通信模块 2 10H: 以太网通信模块 1 11H: 以太网通信模块 2 12H: 以太网通信模块 3 13H: 以太网通信模块 4 14H: 以太网通信模块 5 15H: 以太网通信模块 6 16H: 以太网通信模块 7 17H: 以太网通信模块 8	
	6H: 充电桩	00H: 充电控制	
	7H: 有序用电	00H: 台区负荷预测及控制 01H: 订单状态 02H: 充电服务 03H: 路由器信息 04H: 路由器白名单 05H: 路由器密钥信息表 06H: 路由器密钥信息单元 07H: 时段负荷单元 08H: 充电计划单单元 09H: 充电申请单单元 0AH: 邀约对象单元	
	8H: 负荷感知	00H: 电器设备集合 01H: 电器设备当前数据单元 02H: 电器设备冻结数据单元 03H: 电器设备采样数据单元	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
4H: 参变量	8H: 负荷感知	04H: 计量配置信息	
	9H: TU 设备	00H: 监控单元集合 01H: 死区参数 02H: PT 参数 03H: CT 参数 04H: 低电压报警 05H: 过电压报警 06H: 重载报警 07H: 过载报警 08H: 开入量采集防抖时间 09H: 分闸输出脉冲保持时间 0AH: 蓄电池自动活化周期 0BH: 电池电压低报警阈值 0CH: 短路故障报警 0EH: 录波电场下降比例 0FH: 永久故障 10H: 负荷电流超限阈值 11H: 过电流报警阈值 12H: 相线路有电阈值 13H: 故障指示灯自动复归 14H: 故障遥信保持时间 15H: 过流停电跳闸投退 16H: 过流参数 17H: 重合闸 18H: 遮断电流闭锁 19H: 就地重合器模式参数	

8.3.7 OIA1=5H

OIA1=5H的对象标识定义见表225。

表 227 OIA1=5H 对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
5H: 冻结	0H: 通用	00H: 瞬时冻结 01H: 秒冻结 02H: 分钟冻结 03H: 小时冻结 04H: 日冻结 05H: 结算日冻结 06H: 月冻结	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
5H: 冻结	0H: 通用	07H: 年冻结 08H: 时区表切换冻结 09H: 日时段表切换冻结 0AH: 费率电价切换冻结 0BH: 阶梯切换冻结 11H: 阶梯结算冻结	

8.3.8 OIA1=6H

OIA1=6H的对象标识定义见表226。

表 228 OIA1=6H 对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
6H: 采集	0H: 终端	00H: 采集档案配置表 01H: 采集档案配置单元 02H: 搜表 03H: 一个搜表结果 04H: 一个跨台区结果 05H: 户表新增标识 12H: 任务配置表 13H: 任务配置单元 14H: 普通采集方案集 15H: 普通采集方案 16H: 事件采集方案集 17H: 事件采集方案 18H: 透明方案集 19H: 透明方案 1AH: 透明方案结果集 1BH: 一个透明方案结果 1CH: 上报方案集 1DH: 上报方案 32H: 采集状态集 33H: 一个采集状态 34H: 采集任务监控集 35H: 采集任务监控单元 40H: 采集启动时标 41H: 采集成功时标 42H: 采集存储时标 50H: 线损特征字 51H: 线损数据计算单元	

8.3.9 0IA1=7H

0IA1=7H的对象标识定义见表227。

表 229 0IA1=7H 对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
7H: 集合	0H: 通用	00H: 文件集合 01H: 文件 10H: 脚本集合 11H: 脚本 12H: 脚本执行结果集 13H: 一个脚本执行结果	
	1H: 用户扩展	00H: 变量类集合 01H: 参变量集合	
	2H: 有序用电	00H: 充电计划单集 01H: 充电申请单集 02H: 邀约对象集合	
	3H: 物联管理	00H: 容器结合 01H: 应用集合	

8.3.10 0IA1=8H

0IA1=8H的对象标识定义见表228。

表 230 0IA1=8H 对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
8H: 控制	0H: 通用	00H: 远程控制 01H: 保电 02H: 催费告警 03H: 一般中文信息 04H: 重要中文信息	
	1H: 终端	00H: 终端保安定值 01H: 终端功控时段 02H: 功控告警时间 03H: 时段功控 04H: 厂休控 05H: 营业报停控 06H: 当前功率下浮控 07H: 购电控 08H: 月电控 09H: 时段功控配置单元	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
8H: 控制	1H: 终端	0AH: 厂休控配置单元 0BH: 营业报停控配置单元 0CH: 购电控配置单元 0DH: 月电控配置单元 0EH: 控制对象 0FH: 跳闸轮次 10H: 电控定值	
	2H: TU 设备	00H: 合分遥控 01H: 遥控软压板投退 02H: 电池活化/退出 03H: 信号复归 04H: 解除闭锁	

8.3.11 OIA1=FH

OIA1=FH的对象标识定义见表229。

表 231 OIA1=FH 对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
FH: 其他	0H: 文件传输	00H: 分帧传输管理 01H: 分块传输管理 02H: 扩展传输管理	
	1H: 安全	00H: ESAM 01H: 安全模式参数	
	2H: 输入输出设备	00H: RS232 01H: RS485 02H: 红外 03H: 开关量输入 04H: 直流模拟量 05H: 继电器输出 06H: 告警输出 07H: 多功能端子 08H: 交采接口 09H: 载波/微功率无线接口 0AH: 脉冲输入设备 0BH: 蓝牙 0CH: 230M 无线专网接口 0DH: 模组 0EH: USB 0FH: 服务组件	

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
FH: 其他	2H: 输入输出设备	10H: 从节点单元 12H: 节点相位信息单元 13H: 对象容器集合	
	3H: 显示	00H: 自动轮显 01H: 按键轮显 02H: 图像	
	4H: 物联管理	00H: 日志管理 01H: 容器单元 02H: 应用单元	
	FH: 厂商自定义		

附 录 A
(规范性附录)
对象标识定义

A.1 OIA1 与接口类对象

OIA1与接口类对象的对应关系见A.1 。

表 A.1 OIA1 与接口类对象的对应关系

接口类对象	OIA1									
	0H	1H	2H	3H	4H	5H	6H	7H	8H	FH
电能量类	√									
最大需量类		√	√							
分相变量类			√							
功率类			√							
谐波变量类			√							
数据变量类			√							
事件类				√						
参变量类			√	√	√		√	√	√	√
冻结类						√				
采集类							√			
集合类							√	√	√	
脉冲计量类			√							
控制类									√	
区间统计类			√							
累加平均类			√							
极值统计类			√							
显示类										√
文件传输类										√
设备管理类					√					
应用连接类					√					
ESAM 接口类										√
输入输出设备类										√
总加组类			√							
分项事件类				√						
无线公网通信接口类					√					
以太网通信接口类					√					
厂商自定义类										√

A.2 0IA1=0H

0IA1=0H的对象标识定义见表A.2。

表 A.2 0IA1=0H 对象标识定义

OI	IC	对象名称	实例的对象属性及方法定义
0000	1	组合有功电能	电能量::=double-long; 单位: kWh, 换算: -2 扩展精度电能量::=long64; 单位: kWh, 换算: -4
0010	1	正向有功电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0011	1	A相正向有功电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0012	1	B相正向有功电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0013	1	C相正向有功电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0020	1	反向有功电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0021	1	A相反向有功电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0022	1	B相反向有功电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0023	1	C相反向有功电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0030	1	组合无功1电能	电能量::=double-long; 单位: kvarh, 换算: -2 扩展精度电能量::=long64; 单位: kvarh, 换算: -4
0031	1	A相组合无功1电能	电能量::=double-long; 单位: kvarh, 换算: -2 扩展精度电能量::=long64; 单位: kvarh, 换算: -4
0032	1	B相组合无功1电能	电能量::=double-long; 单位: kvarh, 换算: -2 扩展精度电能量::=long64; 单位: kvarh, 换算: -4
0033	1	C相组合无功1电能	电能量::=double-long; 单位: kvarh, 换算: -2 扩展精度电能量::=long64; 单位: kvarh, 换算: -4
0040	1	组合无功2电能	电能量::=double-long; 单位: kvarh, 换算: -2 扩展精度电能量::=long64; 单位: kvarh, 换算: -4
0041	1	A相组合无功2电能	电能量::=double-long; 单位: kvarh, 换算: -2 扩展精度电能量::=long64; 单位: kvarh, 换算: -4
0042	1	B相组合无功2电能	电能量::=double-long; 单位: kvarh, 换算: -2 扩展精度电能量::=long64; 单位: kvarh, 换算: -4
0043	1	C相组合无功2电能	电能量::=double-long; 单位: kvarh, 换算: -2 扩展精度电能量::=long64; 单位: kvarh, 换算: -4
0050	1	第一象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4

OI	IC	对象名称	实例的对象属性及方法定义
0051	1	A相第一象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0052	1	B相第一象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0053	1	C相第一象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0060	1	第二象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0061	1	A相第二象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0062	1	B相第二象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0063	1	C相第二象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0070	1	第三象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0071	1	A相第三象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0072	1	B相第三象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0073	1	C相第三象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0080	1	第四象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0081	1	A相第四象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0082	1	B相第四象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0083	1	C相第四象限无功电能	电能量::=double-long-unsigned; 单位: kvarh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kvarh, 换算: -4
0090	1	正向视在电能	电能量::=double-long-unsigned; 单位: kVAh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kVAh, 换算: -4
0091	1	A相正向视在电能	电能量::=double-long-unsigned; 单位: kVAh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kVAh, 换算: -4
0092	1	B相正向视在电能	电能量::=double-long-unsigned; 单位: kVAh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kVAh, 换算: -4
0093	1	C相正向视在电能	电能量::=double-long-unsigned; 单位: kVAh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kVAh, 换算: -4
00A1	1	A相反向视在电能	电能量::=double-long-unsigned; 单位: kVAh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kVAh, 换算: -4

OI	IC	对象名称	实例的对象属性及方法定义
00A2	1	B 相反向视在电能	电能量::=double-long-unsigned; 单位: kVAh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kVAh, 换算: -4
00A3	1	C 相反向视在电能	电能量::=double-long-unsigned; 单位: kVAh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kVAh, 换算: -4
0110	1	正向有功基波总电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0111	1	A 相正向有功基波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0112	1	B 相正向有功基波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0113	1	C 相正向有功基波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0120	1	反向有功基波总电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0121	1	A 相反向有功基波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0122	1	B 相反向有功基波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0123	1	C 相反向有功基波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0210	1	正向有功谐波总电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0211	1	A 相正向有功谐波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0212	1	B 相正向有功谐波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0213	1	C 相正向有功谐波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0220	1	反向有功谐波总电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0221	1	A 相反向有功谐波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0222	1	B 相反向有功谐波电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0300	1	铜损有功总电能补偿量	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0301	1	A 相铜损有功电能补偿量	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0302	1	B 相铜损有功电能补偿量	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4

OI	IC	对象名称	实例的对象属性及方法定义
0303	1	C 相铜损有功电能补偿量	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0400	1	铁损有功总电能补偿量	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0401	1	A 相铁损有功电能补偿量	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0402	1	B 相铁损有功电能补偿量	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0403	1	C 相铁损有功电能补偿量	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0500	1	关联总电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0501	1	A 相关联电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0502	1	B 相关联电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4
0503	1	C 相关联电能	电能量::=double-long-unsigned; 单位: kWh, 换算: -2 扩展精度电能量::=long64-unsigned; 单位: kWh, 换算: -4

A.3 OIA1=1H

OIA1=1H的对象标识定义见表A.3。

表A.3 OIA1=1H对象标识定义

OI	IC	对象名称	实例的对象属性及方法定义
1010	2	正向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1011	2	A相正向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1012	2	B相正向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1013	2	C相正向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1020	2	反向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1021	2	A相反向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1022	2	B相反向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1023	2	C相反向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1030	2	组合无功1最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1031	2	A相组合无功1最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1032	2	B相组合无功1最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1033	2	C相组合无功1最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1040	2	组合无功2最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1041	2	A相组合无功2最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1042	2	B相组合无功2最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1043	2	C相组合无功2最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1050	2	第一象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1051	2	A相第一象限最大需量	最大需量值::=double-long-unsigned

OI	IC	对象名称	实例的对象属性及方法定义
			单位: kvar, 换算: -4
1052	2	B相第一象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1053	2	C相第一象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1060	2	第二象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1061	2	A相第二象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1062	2	B相第二象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1063	2	C相第二象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1070	2	第三象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1071	2	A相第三象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1072	2	B相第三象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1073	2	C相第三象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1080	2	第四象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1081	2	A相第四象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1082	2	B相第四象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1083	2	C相第四象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1090	2	正向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
1091	2	A相正向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
1092	2	B相正向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
1093	2	C相正向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
10A0	2	反向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
10A1	2	A相反向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4

OI	IC	对象名称	实例的对象属性及方法定义
10A2	2	B 相反向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
10A3	2	C 相反向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
1110	2	冻结周期内正向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1111	2	冻结周期内 A 相正向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1112	2	冻结周期内 B 相正向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1113	2	冻结周期内 C 相正向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1120	2	冻结周期内反向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1121	2	冻结周期内 A 相反向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1122	2	冻结周期内 B 相反向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1123	2	冻结周期内 C 相反向有功最大需量	最大需量值::=double-long-unsigned 单位: kW, 换算: -4
1130	2	冻结周期内组合无功 1 最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1131	2	冻结周期内 A 相组合无功 1 最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1132	2	冻结周期内 B 相组合无功 1 最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1133	2	冻结周期内 C 相组合无功 1 最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1140	2	冻结周期内组合无功 2 最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1141	2	冻结周期内 A 相组合无功 2 最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1142	2	冻结周期内 B 相组合无功 2 最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1143	2	冻结周期内 C 相组合无功 2 最大需量	最大需量值::=double-long 单位: kvar, 换算: -4
1150	2	冻结周期内第一象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1151	2	冻结周期内 A 相第一象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1152	2	冻结周期内 B 相第一象限最大需量	最大需量值::=double-long-unsigned

OI	IC	对象名称	实例的对象属性及方法定义
			单位: kvar, 换算: -4
1153	2	冻结周期内 C 相第一象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1160	2	冻结周期内第二象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1161	2	冻结周期内 A 相第二象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1162	2	冻结周期内 B 相第二象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1163	2	冻结周期内 C 相第二象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1170	2	冻结周期内第三象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1171	2	冻结周期内 A 相第三象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1172	2	冻结周期内 B 相第三象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1173	2	冻结周期内 C 相第三象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1180	2	冻结周期内第四象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1181	2	冻结周期内 A 相第四象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1182	2	冻结周期内 B 相第四象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1183	2	冻结周期内 C 相第四象限最大需量	最大需量值::=double-long-unsigned 单位: kvar, 换算: -4
1190	2	冻结周期内正向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
1191	2	冻结周期内 A 相正向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
1192	2	冻结周期内 B 相正向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
1193	2	冻结周期内 C 相正向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
11A0	2	冻结周期内反向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
11A1	2	冻结周期内 A 相反向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4
11A2	2	冻结周期内 B 相反向视在最大需量	最大需量值::=double-long-unsigned 单位: kVA, 换算: -4

OI	IC	对象名称	实例的对象属性及方法定义
11A3	2	冻结周期内 C 相反向视在最大需量	最大需量值:=double-long-unsigned 单位: kVA, 换算: -4
注: 组合无功最大需量的正负表示最大需量的潮流方向, 其值是从参与组合无功电能运算的象限中抽取的最大值。			

A.4 OIA1=2H

OIA1=2H的对象标识定义见表A.4。

表 A.4 OIA1=2H 对象标识定义

OI	IC	对象名称	实例的对象属性及方法定义
2000	3	电压	数据类型: long-unsigned, 单位: V, 换算: -1
2001	3	电流	数据类型: double-long, 单位: A, 换算: -3 属性 4 (零线电流) ::=double-long, 单位: A, 换算: -3 属性 5 (剩余电流) ::=double-long, 单位: A, 换算: -3
2002	3	电压相角	数据类型: long-unsigned, 单位: 度, 换算: -1
2003	3	电压电流相角	数据类型: long-unsigned, 单位: 度, 换算: -1
2004	4	有功功率	数据类型: double-long, 单位: W, 换算: -1
2005	4	无功功率	数据类型: double-long, 单位: var, 换算: -1
2006	4	视在功率	数据类型: double-long, 单位: VA, 换算: -1
2007	4	一分钟平均有功功率	数据类型: double-long, 单位: W, 换算: -1
2008	4	一分钟平均无功功率	数据类型: double-long, 单位: var, 换算: -1
2009	4	一分钟平均视在功率	数据类型: double-long, 单位: VA, 换算: -1
200A	4	功率因数	数据类型: long, 单位: 无, 换算: -3
200B	3	电压波形失真度	数据类型: long, 单位: %, 换算: -2
200C	3	电流波形失真度	数据类型: long, 单位: %, 换算: -2
200D	5	电压谐波含有率 (总及 2…n 次)	数据类型: long, 单位: %, 换算: -2
200E	5	电流谐波含有率 (总及 2…n 次)	数据类型: long, 单位: %, 换算: -2
200F	6	电网频率	数据类型: long-unsigned, 单位: Hz, 换算: -2
2010	6	表内温度	属性 2 ::=long (单位: °C, 换算: -1) 属性 4 (端子温度) ::=array long (单位: °C, 换算: -1)
2011	6	时钟电池电压	数据类型: long-unsigned, 单位: V, 换算: -2
2012	6	停电抄表电池电压	数据类型: long-unsigned, 单位: V, 换算: -2
2013	6	时钟电池工作时间	数据类型: double-long-unsigned, 单位: 分钟, 无换算
2014	6	电能表运行状态字	属性 2 (电能表运行状态字) ::= array bit-string, 无单位, 无换算, 包括电能表运行状态字 1…7, 见附录 G 属性 4 (计量模组系统状态字) ::= bit-string(SIZE(32)), 无单位, 无换算 属性 5 (采样系统状态字) ::= bit-string(SIZE(32)), 无单位, 无换算 方法 127 (确认计量模组系统状态字) ::= bit-string(SIZE(32))
2015	6	电能表跟随上报状态字	数据类型: bit-string(SIZE(32)), 无单位, 无换算, 见附录 G 属性 4 (电能表跟随上报模式字) ::= bit-string(SIZE(32)) 方法 127 (确认电能表跟随上报状态字) ::= bit-string(SIZE(32))

OI	IC	对象名称	实例的对象属性及方法定义
2017	6	当前有功需量	数据类型: double-long, 单位: kW, 换算: -4
2018	6	当前无功需量	数据类型: double-long, 单位: kvar, 换算: -4
2019	6	当前视在需量	数据类型: double-long, 单位: kVA, 换算: -4
201A	6	当前电价	数据类型: double-long-unsigned 单位: 元/kWh, 换算: -4
201B	6	当前费率电价	数据类型: double-long-unsigned 单位: 元/kWh, 换算: -4
201C	6	当前阶梯电价	数据类型: double-long-unsigned 单位: 元/kWh, 换算: -4
201E	8	事件发生时间	数据类型: date_time_s
2020	8	事件结束时间	数据类型: date_time_s
2021	8	数据冻结时间	数据类型: date_time_s
2022	8	事件记录序号	数据类型: double-long-unsigned
2023	8	冻结记录序号	数据类型: double-long-unsigned
2024	8	事件发生源	具体对象定义。
2025	8	事件当前值	structure { 事件发生次数 double-long-unsigned, 事件累计时间 double-long-unsigned (单位: 秒, 无换算) }
2026	6	电压不平衡率	数据类型: long-unsigned, 单位: %, 换算: -2
2027	6	电流不平衡率	数据类型: long-unsigned, 单位: %, 换算: -2
2028	6	负载率	数据类型: long-unsigned, 单位: %, 换算: -2
2029	6	安时值	属性 2 安时数值::=array 相安时值 相安时值::=double-long-unsigned, 单位: Ah, 换算: -2 相安时值包括总、A、B、C 相, 总为各相安时值算术和。
202A	8	目标服务器地址	属性 2::=TSA
202C	8	(当前) 钱包文件	数值::=structure { 剩余金额 double-long-unsigned (单位: 元, 换算: -2), 购电次数 double-long-unsigned }
202D	6	(当前) 透支金额	数据类型: double-long-unsigned, 单位: 元, 换算: -2
202E	6	累计购电金额	数据类型: double-long-unsigned, 单位: 元, 换算: -2
2031	6	月度用电量	属性 2 用电量::=double-long-unsigned, 单位: kWh, 换算: -2
2032	6	阶梯结算用电量	属性 2 用电量::=double-long-unsigned, 单位: kWh, 换算: -2
2040	6	控制命令执行状态字	数据类型: bit-string(SIZE(16)), 无单位, 无换算
2041	6	控制命令错误状态字	数据类型: bit-string(SIZE(16)), 无单位, 无换算
2042	6	外置蓝牙负荷开关状态	数据类型: bit-string(SIZE(8)), 无单位, 无换算

OI	IC	对象名称	实例的对象属性及方法定义
			bit0: 继电器命令状态 (0 合闸, 1 跳闸) bit1: 继电器实际状态 (0 合闸, 1 跳闸) bit2: 继电器合闸允许状态 (0 无效, 1 合闸允许) 如果 bit2=1, 则 bit1、bit0 状态无意义。
2043	6	端子座温度分钟变化量	属性 2 (端子温度分钟变化量) ::=array long (单位: °C, 换算: -1)
2044	3	自监测误差	属性 2 ::= array long (单位: %, 换算: -2)
2046	5	谐波电压	数据类型: long-unsigned, 单位: V, 换算: -1
2047	5	谐波电流	数据类型: double-long, 单位: A, 换算: -3
2048	5	谐波有功功率	数据类型: double-long, 单位: W, 换算: -1
2050	6	电流回路状态	属性 2 ::=array enum { 正常 (0), 开路 (1), 短路 (2), 插接半导体 (3) }
205E	6	停电期间数据	属性 2 ::=structure { 停电发生时间 date_time_s, 停电期间电压 array long-unsigned (单位: V, 换算: -1), 停电期间电流 array double-long (单位: A, 换算: -3) }
2100	14	分钟区间统计	统计周期单位为分钟
2101	14	小时区间统计	统计周期单位为小时
2102	14	日区间统计	统计周期单位为日
2103	14	月区间统计	统计周期单位为月
2104	14	年区间统计	统计周期单位为年
2110	15	分钟累加平均	统计周期单位为分钟
2111	15	小时累加平均	统计周期单位为时
2112	15	日累加平均	统计周期单位为日
2113	15	月累加平均	统计周期单位为月
2114	15	年累加平均	统计周期单位为年
2120	16	分钟极值统计	统计周期单位为分钟
2121	16	小时极值统计	统计周期单位为时
2122	16	日极值统计	统计周期单位为日
2123	16	月极值统计	统计周期单位为月
2124	16	年极值统计	统计周期单位为年
2131	6	A 相电压合格率	属性 2 (电压合格率数据) ::=structure { 当日电压合格率 电压合格率, }

OI	IC	对象名称	实例的对象属性及方法定义
			当月电压合格率 电压合格率 } 电压合格率::=structure { 电压监测时间 double-long-unsigned (单位: 分钟, 无换算), 电压合格率 long-unsigned (单位: %, 换算: -2), 电压超限率 long-unsigned (单位: %, 换算: -2), 电压超上限时间 double-long-unsigned (单位: 分钟, 无换算), 电压超下限时间 double-long-unsigned (单位: 分钟, 无换算) }
2132	6	B相电压合格率	同 2131
2133	6	C相电压合格率	同 2131
2140	2	日最大有功功率及发生时间	最大功率及发生时间::=structure { 最大功率值 double-long-unsigned, 发生时间 date_time_s } 功率单位: kW, 换算: -4
2141	2	月最大有功功率及发生时间	同 2140。
2200	6	通信流量	数值::=structure { 当日通信流量 double-long-unsigned, 当月通信流量 double-long-unsigned } 单位: byte, 换算: 0
2203	6	供电时间	数值::=structure { 日供电累计时间 double-long-unsigned, 月供电累计时间 double-long-unsigned } 单位: 分钟, 换算: 0
2204	6	复位次数	数值::=structure { 日复位累计次数 long-unsigned, 月复位累计次数 long-unsigned } 单位: 无, 换算: 0
2301	23	总加组 1	
2302	23	总加组 2	
2303	23	总加组 3	
2304	23	总加组 4	

OI	IC	对象名称	实例的对象属性及方法定义
2305	23	总加组 5	
2306	23	总加组 6	
2307	23	总加组 7	
2308	23	总加组 8	
2401	12	脉冲计量 1	
2402	12	脉冲计量 2	
2403	12	脉冲计量 3	
2404	12	脉冲计量 4	
2405	12	脉冲计量 5	
2406	12	脉冲计量 6	
2407	12	脉冲计量 7	
2408	12	脉冲计量 8	
2500	6	累计水（热）流量	数据类型：double-long-unsigned，单位：m3，换算：-4
2501	6	累计气流量	数据类型：double-long-unsigned，单位：m3，换算：-4
2502	6	累计热量	数据类型：double-long-unsigned，单位：J，换算：-2
2503	6	热功率	数据类型：double-long-unsigned，单位：J/h，换算：-2
2504	6	累计工作时间	数据类型：double-long-unsigned，单位：小时，换算：0
2505	6	水温	数值::=structure { 供水温度 double-long-unsigned, 回水温度 double-long-unsigned } 单位：℃，换算：-2。
2506	6	（仪表）状态 ST	数值::=structure { 阀门状态 enum{开(0)，关(1)，保留(2)，异常(3)}， 电池电压 enum{正常(0)，欠压(0)} }
2600	6	监控单元故障状态字	数值::=bit-string 单位：无，换算：无
2601	6	监控单元运行状态字	数值::=bit-string 单位：无，换算：无
2700	3	充电输出电压	数据类型：long-unsigned 单位：V，换算：-1
2701	3	充电输出电流	数据类型：double-long 单位：A 换算：-3
2702	6	充电导引电压	数据类型：long-unsigned 单位：V，换算：-1
2703	6	输出 PWM 占空比	数据类型：long-unsigned 单位：%，换算：-2

OI	IC	对象名称	实例的对象属性及方法定义
2704	6	充电接口温度探头温度	数据类型: array long 单位: °C, 换算: -1
2705	6	SOC	数据类型: long-unsigned 单位: %, 换算: 0
2706	6	电池	属性 2 (电池组最低温度) ::= long (单位: °C, 换算: -1) 属性 4 (电池组最高温度) ::= long (单位: °C, 换算: -1) 属性 5 (单体电池最低温度) ::= long (单位: °C, 换算: -1) 属性 6 (单体电池最高温度) ::= long (单位: °C, 换算: -1)
2707	6	充电机环境温度	数据类型 ::= long 单位: V, 换算: -1
2708	6	BMS	属性 2 (需求电压) ::= long-unsigned (单位: V, 换算: -1) 属性 4 (需求电流) ::= double-long (单位: A 换算: -3) 属性 5 (充电模式) ::= enum { 恒压充电 (1), 恒流充电 (2) } 属性 6 (充电电压测量) ::= long-unsigned (单位: V, 换算: -1) 属性 7 (充电电流测量值) ::= double-long (单位: A 换算: -3)
2709	6	当前充电状态	属性 2 ::= structure { 估算剩余充电时间 long-unsigned (单位: 分钟, 换算: 0), 当前充电电量 double-long-unsigned (单位: kWh, 换算: -2), 累计充电时长 long-unsigned, (单位: 分钟, 换算: 0) }
<p>注 1: 三相三线电能表电压 A 相为 U_{ab}, B 相为 0, C 相为 U_{cb}; 电流 A 相为 I_a, B 相为 0, C 相为 I_c; 功率因数 A 相为 U_{ab} 与 I_a 的夹角余弦, B 相为 0, C 相为 U_{cb} 与 I_c 的夹角余弦。</p> <p>注 2: 电流、有功功率、无功功率、当前有功需量、当前无功需量、当前视在需量按潮流方向分为正负数, 正数代表输入, 负数代表输出。表内温度正数为摄氏零上, 负数为摄氏零下。</p> <p>注 3: 相角测量范围是 0...360 度。</p> <p>注 4: 当前有功需量、当前无功需量、当前视在需量是最近一段时间的滑差功率。</p> <p>注 5: 组合无功最大需量的正负数是用来标志潮流的方向, 组合无功最大需量从参与组合电能运算的象限中抽取最大值, 如果来自象限 3、4, 以负数表示。</p>			

A.5 0IA1=3H

0IA1=3H的对象标识定义见表A.5。

表 A.5 0IA1=3H 对象标识定义

OI	IC	对象名称	实例的对象属性及方法定义
3000	24	电能表失压事件	属性 5（配置参数） ::= structure { 电压触发上限 long-unsigned（单位：V，换算：-1）， 电压恢复下限 long-unsigned（单位：V，换算：-1）， 电流触发下限 double-long（单位：A，换算：-4）， 判定延时时间 unsigned（单位：s，换算：0） } 属性 13（失压统计） ::= structure { 事件发生总次数 double-long-unsigned， 事件总累计时间 double-long-unsigned（单位：秒，无换算）， 最近一次失压发生时间 date_time_s， 最近一次失压结束时间 date_time_s } 事件发生总次数：A、B、C 相失压累计次数之和。 事件总累计时间：A、B、C 相失压累计时间之和。
3001	24	电能表欠压事件	属性 5（配置参数） ::= structure { 电压触发上限 long-unsigned（单位：V，换算：-1）， 判定延时时间 unsigned（单位：s，换算：0） }
3002	24	电能表过压事件	属性 5（配置参数） ::= structure { 电压触发下限 long-unsigned（单位：V，换算：-1）， 判定延时时间 unsigned（单位：s，换算：0） }
3003	24	电能表断相事件	属性 5（配置参数） ::= structure { 电压触发上限 long-unsigned（单位：V，换算：-1）， 电流触发上限 double-long（单位：A，换算：-4）， 判定延时时间 unsigned（单位：s，换算：0） }
3004	24	电能表失流事件	属性 5（配置参数） ::= structure { 电压触发下限 long-unsigned（单位：V，换算：-1），

OI	IC	对象名称	实例的对象属性及方法定义
			电流触发上限 double-long (单位: A, 换算: -4), 电流触发下限 double-long (单位: A, 换算: -4), 判定延时时间 unsigned (单位: s, 换算: 0) }
3005	24	电能表过流事件	属性 5 (配置参数) ::=structure { 电流触发限值 double-long (单位: A, 换算: -4), 判定延时时间 unsigned (单位: s, 换算: 0) }
3006	24	电能表断流事件	属性 5 (配置参数) ::=structure { 电压触发下限 long-unsigned (单位: V, 换算: -1), 电流触发上限 double-long (单位: A, 换算: -4), 判定延时时间 unsigned (单位: s, 换算: 0) }
3007	24	电能表功率反向事件	属性 5 (配置参数) ::=structure { 有功功率触发限值 double-long (单位: W, 换算: -1), 判定延时时间 unsigned (单位: s, 换算: 0) }
3008	24	电能表过载事件	属性 5 (配置参数) ::=structure { 有功功率触发限值 double-long (单位: W, 换算: -1), 判定延时时间 unsigned (单位: s, 换算: 0) }
3009	7	电能表正向有功需量超限事件	属性 2 (事件记录表) ::=array 电能表需量超限事件单元 属性 6 (配置参数) ::=structure { 触发限值 double-long-unsigned (单位: kW, 换算: -4), 判定延时时间 unsigned (单位: s, 换算: 0) } 事件发生源::=NULL
300A	7	电能表反向有功需量超限事件	属性 2 (事件记录表) ::=array 电能表需量超限事件单元 属性 6 (配置参数) ::=structure { 触发限值 double-long-unsigned (单位: kW, 换算: -4), 判定延时时间 unsigned (单位: s, 换算: 0) } 事件发生源::=NULL
300B	24	电能表无功需量超限事件	属性 5 (配置参数) ::=structure {

OI	IC	对象名称	实例的对象属性及方法定义
			触发限值 double-long-unsigned (单位: kvar, 换算: -4), 判定延时时间 unsigned (单位: s, 换算: 0) } 属性 6 (事件记录表 1) ::=array 电能表需量超限事件单元 属性 7 (事件记录表 2) ::=array 电能表需量超限事件单元 属性 8 (事件记录表 3) ::=array 电能表需量超限事件单元 属性 9 (事件记录表 4) ::=array 电能表需量超限事件单元
300C	7	电能表功率因数超下限事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 下限阈值 long (单位: %, 换算: -1), 判定延时时间 unsigned (单位: s, 换算: 0) } 事件发生源::=NULL
300D	7	电能表全失压事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { } 事件发生源::=NULL
300E	7	电能表辅助电源掉电事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 判定延时 unsigned (单位: s, 换算: 0) } 事件发生源::=NULL
300F	7	电能表电压逆相序事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 判定延时 unsigned (单位: s, 换算: 0) } 事件发生源::=NULL
3010	7	电能表电流逆相序事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 判定延时 unsigned (单位: s, 换算: 0) } 事件发生源::=NULL
3011	7	电能表掉电事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 判定延时 unsigned (单位: s, 换算: 0) }

OI	IC	对象名称	实例的对象属性及方法定义
			事件发生源::=NULL
3012	7	电能表编程事件	属性 2（事件记录表）::=array 编程记录事件单元 属性 6（配置参数）::=structure { }
3013	7	电能表清零事件	属性 2（事件记录表）::=array 标准事件记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
3014	7	电能表需量清零事件	属性 2（事件记录表）::=array 标准事件记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
3015	7	电能表事件清零事件	属性 2（事件记录表）::=array 事件清零事件记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
3016	7	电能表校时事件	属性 2（事件记录表）::=array 标准事件记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
3017	7	电能表时段表编程事件	属性 2（事件记录表）::=array 电能表时段表编程事件记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
3018	7	电能表时区表编程事件	属性 2（事件记录表）::=array 标准事件记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
3019	7	电能表周日编程事件	属性 2（事件记录表）::=array 标准事件记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
301A	7	电能表结算日编程事件	属性 2（事件记录表）::=array 标准事件记录单元

OI	IC	对象名称	实例的对象属性及方法定义
			属性 6（配置参数） ::= structure { } 事件发生源 ::= NULL
301B	7	电能表开盖事件	属性 2（事件记录表） ::= array 标准事件记录单元 属性 6（配置参数） ::= structure { } 事件发生源 ::= NULL
301C	7	电能表开端钮盒事件	属性 2（事件记录表） ::= array 标准事件记录单元 属性 6（配置参数） ::= structure { } 事件发生源 ::= NULL
301D	7	电能表电压不平衡事件	属性 2（事件记录表） ::= array 标准事件记录单元 属性 6（配置参数） ::= structure { 限值 long（单位：%，换算：-2）， 判定延时时间 unsigned（单位：s，换算：0） } 事件发生源 ::= NULL
301E	7	电能表电流不平衡事件	属性 2（事件记录表） ::= array 标准事件记录单元 属性 6（配置参数） ::= structure { 限值 long（单位：%，换算：-2）， 判定延时时间 unsigned（单位：s，换算：0） } 事件发生源 ::= NULL
301F	7	电能表跳闸事件	属性 2（事件记录表） ::= array 标准事件记录单元 属性 6（配置参数） ::= structure { 端子过热跳闸温度阈值 long（单位：℃，换算：-1）， 端子过热跳闸恢复温度阈值 long（单位：℃，换算：-1）， 端子座过热跳闸电流阈值 double-long-unsigned（单位：A， 换算：-4）， 端子过热跳闸延时时间 unsigned（单位：s，换算：0） } 事件发生源 ::= enum { 远程控制跳闸 (0)， 本地费控跳闸 (1)， }

OI	IC	对象名称	实例的对象属性及方法定义
			端子座过热跳闸 (2) 过载 (3), 用户行为 (4) }
3020	7	电能表合闸事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { } 事件发生源::=enum { 远程控制合闸 (0), 本地费控合闸 (1), 端子座过热恢复后合闸 (2), 过载恢复后合闸 (3), 用户行为 (4) } 注: 过载恢复后合闸由管理模组给负荷开关发合闸允许命令, 经人工合闸, 端子座过热恢复后合闸事件以外置负荷开关状态变化记录。
3021	7	电能表节假日编程事件	属性 2 (事件记录表) ::=array 电能表节假日编程事件单元 属性 6 (配置参数) ::=structure { } 事件发生源::=NULL
3022	7	电能表有功组合方式编程事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { } 事件发生源::=NULL
3023	7	电能表无功组合方式编程事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { } 事件发生源::=enum { 无功组合方式 1 特征字 (0), 无功组合方式 2 特征字 (1) }
3024	7	电能表费率参数表编程事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure {

OI	IC	对象名称	实例的对象属性及方法定义
			} 事件发生源::=NULL
3025	7	电能表阶梯表编程事件	属性 2（事件记录表）::=array 标准事件记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
3026	7	电能表密钥更新事件	属性 2（事件记录表）::=array 标准事件记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
3027	7	电能表异常插卡事件	属性 2（事件记录表）::=array 电能表异常插卡记录单元 属性 6（配置参数）::=structure { } 属性 11（非法插卡总次数）::=double-long-unsigned 事件发生源::=NULL
3028	7	电能表购电记录	属性 2（事件记录表）::=array 标准事件记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
3029	7	电能表退费记录	属性 2（事件记录表）::=array 电能表退费记录单元 属性 6（配置参数）::=structure { } 事件发生源::=NULL
302A	7	电能表恒定磁场干扰事件	属性 2（事件记录表）::=array 标准事件记录单元 事件发生源::=NULL
302B	7	电能表负荷开关误动作事件	属性 2（事件记录表）::=array 标准事件记录单元 事件发生源::=NULL
302C	7	电能表电源异常事件	属性 2（事件记录表）::=array 标准事件记录单元 事件发生源::=NULL
302D	7	电能表电流严重不平衡事件	属性 2（事件记录表）::=array 标准事件记录单元 属性 6（配置参数）::=structure { 限值 long（单位：%，换算：-2） 判定延时时间 unsigned（单位：s，换算：0） } 事件发生源::=NULL

OI	IC	对象名称	实例的对象属性及方法定义
302E	7	电能表时钟故障事件	属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::=structure { } 事件发生源 ::=NULL
302F	7	电能表计量芯片故障事件	属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::=structure { 判定延时 unsigned（单位：s，换算：0） } 事件发生源 ::=NULL
3030	7	通信模块变更事件	属性 2（事件记录表） ::=array 通信模块变更事件单元 属性 6（配置参数） ::=structure { 判定延时 unsigned（单位：s，换算：0） } 事件发生源 ::=OAD 事件发生源为通信模块 OAD。
3031	7	管理模组插拔事件	属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::=structure { } 事件发生源 ::=NULL
3032	7	端子座过热报警事件	属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::=structure { 报警温度阈值 long（单位：℃，换算：-1）， 报警恢复温度阈值 long（单位：℃，换算：-1）， 报警延时时间 unsigned（单位：s，换算：0） }
3033	7	端子座温度剧变事件	属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::= structure { 报警温度分钟变化量 long-unsigned（单位：℃，换算：-1） }
3034	7	端子座温度不平衡事件	属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::=structure { 温差超限报警阈值 long（单位：℃，换算：-1）， 温差恢复解除报警阈值 long（单位：℃，换算：-1）， 延时判定时间 unsigned（单位：s，换算：0） }

OI	IC	对象名称	实例的对象属性及方法定义
			}
3035	7	误差自监测超限事件	属性 2（事件记录表） ::= array 标准事件记录单元 属性 6（配置参数） ::= structure { 误差触发下限 long-unsigned（单位：%，换算：-2）， 判定延时时间 unsigned（单位：min，换算：0） } 属性 15（自监测周期） ::= double-long-unsigned（单位：min，换算：0） 误差自监测周期设置为 0 表示不启用误差自监测。
3036	7	升级事件	属性 2（事件记录表） ::= array 标准事件记录单元 方法 127：升级记录同步（参数） 参数 ::= 标准事件记录单元
3037	7	停电事件	属性 2（事件记录表） ::= array 标准事件记录单元
3038	7	上电事件	属性 2（事件记录表） ::= array 标准事件记录单元
3039	24	电压谐波总畸变率超限事件	属性 2（事件记录表） ::= array 标准事件记录单元 属性 5（配置参数） ::= structure { 谐波总畸变率超限阈值 long（单位：%，换算：-2）， 延时判定时间 unsigned（单位：s，换算：0） }
303A	24	电流谐波总畸变率超限事件	属性 2（事件记录表） ::= array 标准事件记录单元 属性 5（配置参数） ::= structure { 谐波总畸变率超限阈值 long（单位：%，换算：-2）， 延时判定时间 unsigned（单位：s，换算：0） }
303B	24	电能表功率因数超下限事件	属性 5（配置参数） ::= structure { 下限阈值 long（单位：%，换算：-1）， 判定延时时间 unsigned（单位：s，换算：0） } 事件发生源 ::= NULL
303C	7	电能表广播校时事件	属性 2（事件记录表） ::= array 标准事件记录单元 属性 6（配置参数） ::= structure { } 事件发生源 ::= NULL
3100	7	终端初始化事件	属性 2（事件记录表） ::= array 标准事件记录单元 属性 6（配置参数） ::= structure {

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> { 关联采集任务号 unsigned } 采集标志: bit0: 置“1”有效, 置“0”无效; bit1: 置“1”随机选择电能表, 置“0”只采集设置的电能表。 事件发生源::=NULL 关联采集任务号: 当关联对象属性 OAD 为 60120300 时, 关联的数据内容为当前配置 的采集任务最近一次数据采集结果集。 </pre>
3107	7	终端直流模拟量越上限事件	<pre> 属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 直流模拟量上限 double-long } 事件发生源::=OAD (直流模拟量号) </pre>
3108	7	终端直流模拟量越下限事件	<pre> 属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 直流模拟量下限 double-long } 事件发生源::=OAD (直流模拟量号) </pre>
3109	7	终端消息认证错误事件	<pre> 属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { } 事件发生源::=NULL </pre>
310A	7	设备故障记录	<pre> 属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { } 事件发生源::=enum { 终端主板内存故障 (0), 时钟故障 (1), 主板通信故障 (2), 485 抄表故障 (3), 显示板故障 (4), 载波通道异常 (5), 内存初始化错误 (6), ESAM 错误 (7), } </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			时钟电池故障 (8), 备用电池故障 (9), 蓝牙模块故障 (10), 计量单元通讯故障 (11), 台区负荷越限 (12), 充电桩通信故障 (13), 充电桩设备故障 (14), 本地订单记录满 (15), 超功率限值充电 (16) }
310B	7	电能表示度下降事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 关联采集任务号 unsigned } 事件发生源::=TSA 采集任务中需要配置相关 OAD 的采集任务。
310C	7	电能量超差事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 阈值 double-long-unsigned (单位: %, 无换算), 关联采集任务号 unsigned } 事件发生源::=TSA 采集任务中需要配置相关 OAD 的采集任务。
310D	7	电能表飞走事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 阈值 double-long-unsigned (单位: %, 无换算), 关联采集任务号 unsigned } 事件发生源::=TSA 采集任务中需要配置相关 OAD 的采集任务。
310E	7	电能表停走事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { 阈值 TI, 关联采集任务号 unsigned } 事件发生源::=TSA 采集任务中需要配置相关 OAD 的采集任务。

OI	IC	对象名称	实例的对象属性及方法定义
310F	7	终端抄表失败事件	属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::=structure { 重试轮次 unsigned, 关联采集任务号 unsigned } 事件发生源 ::=TSA
3110	7	月通信流量超限事件	属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::=structure { 通信流量门限 double-long-unsigned（单位：byte） } 事件发生源 ::=NULL
3111	7	发现未知电能表事件	属性 2（事件记录表） ::=array 发现未知电能表事件单元 属性 6（配置参数） ::=structure { }
3112	7	跨台区电能表事件	属性 2（事件记录表） ::=array 跨台区电能表事件单元 属性 6（配置参数） ::=structure { }
3114	7	终端对时事件	属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::=structure { } 事件发生源 ::=NULL
3115	7	遥控跳闸记录	属性 2（事件记录表） ::=array 遥控事件记录单元 属性 6（配置参数） ::=structure { } 事件发生源 ::=OAD（继电器单元）
3116	7	有功总电能差动超限事件记录	属性 2（事件记录表） ::=array 差动超限事件记录单元 属性 6（配置参数） ::=array 有功总电能差动组配置 有功总电能差动组配置 ::=structure { 有功总电能差动组序号 unsigned, 对比的总加组 OI, 参照的总加组 OI, 参与差动的电能量的时间区间及对比方法标志 unsigned, 差动超限相对偏差值 integer（单位：% ，换算：0）， 差动超限绝对偏差值 long64（单位：kWh，换算：-4） }

OI	IC	对象名称	实例的对象属性及方法定义
			<p>} 参与差动的电能量的时间区间及对比方法标志： bit0…bit1 编码表示电能量的时间跨度，取值范围 0…2 依次表示 60 分钟电量、30 分钟电量、15 分钟电量，其他值无效。 bit2…bit6 备用。 bit7 表示对比方法标志，置“0”：相对对比，公式见公式（1）； 置“1”：绝对对比，公式见公式（2）。 $\frac{ Q-q }{q} \times 100\% \dots\dots (1)$ $Q-q \dots\dots\dots (2)$ 式中：Q 为对比的总加组总电能；q 为参照的总加组总电能。</p>
3117	7	输出回路接入状态变位事件记录	<p>属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::=structure { }</p>
3118	7	终端编程记录	<p>属性 2（事件记录表） ::=array 编程记录事件单元 属性 6（配置参数） ::=structure { }</p>
3119	7	终端电流回路异常事件	<p>属性 2（事件记录表） ::=array 标准记录事件单元 属性 6（配置参数） ::=structure { } 事件发生源 ::=enum{短路(0)，开路(1)}</p>
311A	7	电能表在网状态切换事件	<p>属性 2（事件记录表） ::=array 电能表在网状态切换事件单元 属性 6（配置参数） ::=structure { 判定延时时间 long-unsigned（单位：s，换算：0） } 事件发生源 ::=NULL 此事件只记录电能表在网状态变迁。</p>
311B	7	终端对电能表校时记录	<p>属性 2（事件记录表） ::=array 终端对电能表校时记录单元 属性 6（配置参数） ::=structure { }</p>
311C	7	电能表数据变更监控记录	<p>属性 2（事件记录表） ::=array 电能表数据变更监控记录单元 属性 6（配置参数） ::=structure { 关联采集任务序号 unsigned } 事件发生源 ::=TSA</p>

OI	IC	对象名称	实例的对象属性及方法定义
311D	7	电能表在线状态变化事件	属性 2（事件记录表） ::=array 电能表在线状态变化事件单元 属性 6（配置参数） ::=structure { } 事件发生源 ::=TSA
311E	7	拒绝从节点入网记录	属性 2（事件记录表） ::=array 拒绝从节点入网记录单元 属性 6（配置参数） ::=structure { } 事件发生源 ::=NULL
311F	7	参数变更记录	属性 2（事件记录表） ::=array 参数变更事件记录单元 属性 6（配置参数） ::=structure { } 事件发生源 ::=OAD（参数对象属性描述符）
3120	7	电流互感器异常事件	属性 2（事件记录表） ::=array 电流互感器异常事件单元
3121	7	TA 专用模块工况事件	属性 2（事件记录表） ::=array 标准事件记录单元
3140	7	安全事件变更记录	属性 2（事件记录表） ::=array 安全事件变更记录单元
3141	7	STA 入网事件	属性 2（事件记录表） ::=array STA 入网事件单元
3200	7	功控跳闸记录	属性 2（事件记录表） ::=array 功控跳闸记录单元 属性 6（配置参数） ::=structure { }
3201	7	电控跳闸记录	属性 2（事件记录表） ::=array 电控跳闸记录单元 属性 6（配置参数） ::=structure { }
3202	7	购电参数设置记录	属性 2（事件记录表） ::=array 标准事件记录单元 属性 6（配置参数） ::=structure { } 事件发生源 ::=OI
3203	7	电报告警事件记录	属性 2（事件记录表） ::=array 电报告警事件单元 属性 6（配置参数） ::=structure { }
3300	8	事件上报状态	事件上报状态 ::=array 通道上报状态 通道上报状态 ::=structure { 通道 OAD, 上报状态 unsigned } 上报状态见表 152。
3301	8	标准事件记录单元	标准事件记录单元 ::=structure

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 instance-specific, 事件上报状态 array 通道上报状态, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data } </pre>
3302	8	编程记录事件单元	<pre> 编程记录事件单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 编程对象列表 array OAD, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data } </pre>
3303	8	发现未知电能表事件单元	<pre> 发现未知电能表事件::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 搜表结果 array 一个搜表结果, } </pre>
3304	8	跨台区电能表事件单元	<pre> 跨台区电能表事件单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 跨台区搜表结果 array 一个跨台区结果, 第 1 个关联对象属性的数据 Data, ... } </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			第 n 个关联对象属性的数据 Data }
3305	8	功控跳闸记录单元	功控跳闸记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 OI, 事件上报状态 array 通道上报状态, 事件发生后 2 分钟功率 long64(单位: W, 换算-1), 控制对象 OI, 跳闸轮次 bit-string(SIZE(8)), 功控定值 long64 (单位: kW, 换算-4), 跳闸发生前总加组有功功率 long64 (单位: kW, 换算-4), 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }
3306	8	电控跳闸记录单元	电控跳闸记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 OI, 事件上报状态 array 通道上报状态, 控制对象 OI, 跳闸轮次 bit-string(SIZE(8)), 电控定值 long64 (单位: kWh, 换算-4), 跳闸发生时总加组电能量 long64 (单位: kWh/元, 换算-4), 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data } 说明: 当事件发生源类型为月电控 8108, 跳闸时总加组电能量为总加组月电能量; 当事件发生源类型为购电控 8107, 跳闸时总加组电能量为剩余电能量/费。
3307	8	电控告警事件单元	电控告警事件单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, }

OI	IC	对象名称	实例的对象属性及方法定义
			事件发生源 OI, 事件上报状态 array 通道上报状态, 控制对象 OI, 电控定值 long64 (单位: kWh, 换算-4), 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }
3308	8	电能表需量超限事件单元	电能表需量超限事件单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 超限期间需量最大值 double-long-unsigned, 超限期间需量最大值发生时间 date_time_s, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }
3309	8	停/上电事件记录单元	停/上电事件记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 enum{停电(0), 上电(1)}, 事件上报状态 array 通道上报状态, 属性标志 bit-string (SIZE(8)), 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data } 属性标志: bit0 置“1”: 事件正常, bit0 置“0”: 事件异常; bit1 置“1”: 事件有效, bit1 置“0”: 事件无效; bit2…bit7 备用。
330A	8	遥控事件记录单元	遥控事件记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, }

OI	IC	对象名称	实例的对象属性及方法定义
			事件结束时间 date_time_s, 事件发生源 OAD, 事件上报状态 array 通道上报状态, 控后 2 分钟总加组有功功率 array long64, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }
330B	8	有功总电能差动越限事件记录单元	有功总电能差动越限事件记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 unsigned, 事件上报状态 array 通道上报状态, 越限时对比总加组有功总电能 long64(单位: kWh, 换算: -4), 越限时参照总加组有功总电能 long64(单位: kWh, 换算: -4), 越限时差动越限相对偏差值 integer (单位: %, 换算: 0), 越限时差动越限绝对偏差值 long64 (单位: kWh, 换算: -4) }
330C	8	事件清零事件记录单元	事件清零事件记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 事件清零列表 array OMD }
330D	8	终端对电能表校时记录单元	终端对电能表校时记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 TSA, 事件上报状态 array 通道上报状态, 校时前时钟 date_time_s, 时钟误差 integer (单位: 秒, 无换算) }
330E	8	电能表在网状态切换事件单元	电能表在网状态切换事件单元::=structure {

OI	IC	对象名称	实例的对象属性及方法定义
			事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 状态变迁事件 array structure { 电能表地址 TSA, 在网状态 bool } }
330F	8	电能表数据变更监控记录单元	电能表数据变更监控记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 TSA, 事件上报状态 array 通道上报状态, 监控数据对象 CSD, 变化前数据 Data, 变化后数据 Data } 事件发生时间: 为监控数据发生变化的时刻; 事件结束时间: 无效, 各字段填 FF。
3311	8	退费事件记录单元	退费事件记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 退费金额 double-long-unsigned (单位: 元, 换算: -2), 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }
3312	8	通信模块变更事件单元	通信模块变更事件单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 OAD, }

OI	IC	对象名称	实例的对象属性及方法定义
			事件上报状态 array 通道上报状态, 模块宿主的通信地址 octet-string, 变更前的模块描述符 visible-string, 变更后的模块描述符 visible-string } 注: 模块宿主的通信地址为电能表或采集器通信地址。
3313	8	电能表时钟超差记录单元	电能表时钟超差记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 TSA, 事件上报状态 array 通道上报状态, 电能表时钟 date_time_s, 终端当前时钟 date_time_s, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }
3314	8	电能表时段表编程事件记录单元	电能表时段表编程事件记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 编程时段表对象 OAD, 编程前时段表 日时段表, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data } 日时段表::=array 时段 时段::=structure { 时 unsigned, 分 unsigned, 费率号 unsigned }
3315	8	电能表节假日编程事件记录单元	电能表节假日编程事件记录单元::=structure { 事件记录序号 double-long-unsigned, }

OI	IC	对象名称	实例的对象属性及方法定义
			事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 编程节假日对象 OAD, 编程前节假日内容 公共假日, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data } 公共假日::=structure { 日期 date, 日时段表号 unsigned }
3316	8	安全事件变更记录单元	安全事件变更记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 异常标志 bit-string (SIZE(16)) 以太网远程端口开放信息 array long-unsigned 以太网外联信息 array 外联 IP 和端口 远程通信模块端口开放信息 array long-unsigned 远程通信模块外联信息 array 外联 IP 和端口 USB 设备接入信息 bit-string (SIZE(8)) 本地维护端口信息 bit-string (SIZE(8)) 关键文件目录变更信息 array visible-string 以太网登录控制台信息 bit-string (SIZE(8)) 串口登录控制台信息 bit-string (SIZE(8)) 非法操作命令信息 bit-string (SIZE(8)) } 外联 IP 和端口::=structure { 外联 IP octet-string, 外联端口 long-unsigned }
3317	8	电能表在线状态变化事件单元	电能表在线状态变化事件单元::=structure {

OI	IC	对象名称	实例的对象属性及方法定义
			事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 TSA, 事件上报状态 array 通道上报状态, 状态变化 enum{上线(0), 离线(1)} 离线信息 structure OPTIONAL } 离线信息 ::= structure { 本次离线时常 double-long-unsigned, (单位: 秒) 离线原因 enum{未知(0), 停电(1), 信道变化(2)} } 离线信息仅在状态变化为上线时有效。
3318	8	电流互感器异常事件单元	属性 2 (电流互感器异常事件单元) ::= structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 enum{1:发生, 0:恢复}, 事件上报状态 array 通道上报状态, 异常标志 bit-string(SIZE(8)) } 异常标志: a) D7~D5: 编码表示异常类型, 依次表示 1 代表回路串接半导体、 2 代表一次侧短路、 3 代表 TA 更换变比、 4 代表二次侧短路、 6 代表二次侧开路。 b) D4~D3: 备用。 c) D0~D2: 依次对位表示 A、 B、 C 三相, 置“1”对应相有事件, 置“0”无事件, 可同时有效。
3319	8	监控单元遥信变位事件记录单元	监控单元遥信变位事件记录单元 ::= structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 TSA, 事件上报状态 array 通道上报状态, 当前遥信状态 array 遥信状态, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data } 遥信状态 ::= structure

OI	IC	对象名称	实例的对象属性及方法定义
			<pre>{ 遥信对象 OAD, 遥信值 Data, 变位时间 date_time }</pre>
331A	8	监控单元遥控记录事件单元	<pre>监控单元遥控记录事件单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 TSA, 事件上报状态 array 通道上报状态, 遥控对象 OAD, 遥控命令类型 enum{选择(0), 执行(1), 取消(2)}, 遥控动作类型 enum{分(1), 合(2)}, 第1个关联对象属性的数据 Data, ... 第n个关联对象属性的数据 Data }</pre>
331B	8	拒绝从节点入网记录单元	<pre>拒绝从节点入网记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 TSA, 事件上报状态 array 通道上报状态, 拒绝从节点列表 array octet-string, 第1个关联对象属性的数据 Data, ... 第n个关联对象属性的数据 Data }</pre>
331C	8	参数变更记录单元	<pre>参数变更记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 OAD, 事件上报状态 array 通道上报状态, 参数变更前值 Data, 参数变更后值 Data, 第1个关联对象属性的数据 Data, ... 第n个关联对象属性的数据 Data }</pre>
3320	8	新增上报事件列表	<p>属性 2（新增上报事件列表，只读） ::= array OAD</p> <p>属性 3（需上报事件对象列表，只读） ::= array OI</p> <p>注：新增上报事件列表对象按通信通道（OAD）记录，当该列表中</p>

OI	IC	对象名称	实例的对象属性及方法定义
			的事件记录通过“当前”通道被读取后，从该列表中删除此对象。
3321	8	充电计划生成失败记录单元	充电计划生成失败记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 充电申请单号 octet-string, 路由器资产编号 visible-string (SIZE(22)), 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }
3322	8	充电计划上报记录单元	充电计划上报记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, 充电申请单号 octet-string, 用户 ID visible-string, 决策单元 enum{主站 (1), 控制器 (2)}, 决策类型 enum{生成 (1), 调整 (2)}, 决策时间 date_time_s 路由器资产编号 visible-string (SIZE(22)), 充电需求电量 double-long-unsigned(单位: kWh, 换算: -2), 充电额定功率 double-long (单位: kW, 换算: -4), 充电模式 enum{正常 (0), 有序 (1)} 充电时段 array 时段充电功率 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }
3323	8	充电申请事件记录单元	充电申请事件记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, }

OI	IC	对象名称	实例的对象属性及方法定义
			事件上报状态 array 通道上报状态, 充电申请单号 octet-string, 用户 ID visible-string, 路由器资产编号 visible-string (SIZE(22)), 充电申请时间 date_time_s, 充电需求电量 double-long-unsigned(单位: kWh, 换算: -2), 计划用车时间 date_time_s, 充电方式 enum{正常 (0), 有序 (1)}, 身份认证 Token visible-string (SIZE(32)), 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data }
3324	8	充电执行事件记录单元	充电订单执行事件记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 enum{启动成功(1), 启动失败 (2)}, 事件上报状态 array 通道上报状态, 充电申请单号 octet-string, 电能示值底值 array double-long, 当前电能示值 array double-long, 已充时间 double-long-unsigned(单位: 秒, 换算: 0), 计划充电功率 double-long (单位: kW, 换算: -4), 当前充电功率 double-long (单位: kW, 换算: -4), 电压 array long-unsigned (单位: V, 换算: -1), 电流 array double-long (单位: A 换算: -3), 充电桩状态 enum{待机 (1), 工作 (2), 故障 (3)}, 第 1 个关联对象属性的数据 Data, ... 第 n 个关联对象属性的数据 Data } 电能量::=double-long-unsigned (单位: kWh, 换算: -2)
3325	8	有序充电类事件记录单元	有序充电类事件记录单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 NULL, 事件上报状态 array 通道上报状态, ...

OI	IC	对象名称	实例的对象属性及方法定义
3325	8	有序充电类事件记录单元	<pre> 故障状态 bit-string, 故障变位状态 bit-string } 故障状态对应的 bit 位 { 急停按钮动作故障 (1), 避雷器故障 (2), 充电枪未归位 (3), 过温故障 (4), 输入过压告警 (5), 输入欠压告警 (6), 充电中车辆控制导引告警 (7), 交流接触器故障 (8), 输出过流告警 (9), 输出过流故障 (10), 交流断路器故障 (11), 充电接口电子锁故障 (12), 充电接口过温故障 (13), 充电连接状态 CC 异常 (14), 充电控制状态 CP 异常 (15), PE 断线故障 (16), 门禁故障 (17), 充电桩过温故障 (18), 充电机其他故障 (19) } </pre>
3326	8	充电订单结算记录单元	<pre> 充电订单结算事件记录单元 ::= structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 enum, 事件上报状态 array 通道上报状态, 充电申请单号 octet-string, 用户 ID visible-string, 路由器资产编号 visible-string (SIZE(22)), 充电需求电量 double-long-unsigned (单位: kWh, 换算: -2), 充电申请时间 date_time_s, 计划用车时间 date_time_s, 充电模式 enum{正常 (0), 有序 (1)} 充电启动时电能示值 array double-long } </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			充电结束时电能示值 array double-long 充电启动时间 date_time_s, 充电结束时间 date_time_s, 实际充电电量 array double-long-unsigned, 充电有效时间 double-long-unsigned (单位: 秒, 换算: 0) } 事件发生源::=enum { 正常结束 (0), 用户结束 (1), 台区负荷越限 (2), 路由器故障 (3), 充电桩故障 (4), 系统终止 (5) } 充电有效时间: 实际充电功率不为零的充电累积时间。
3327	8	系统监测事件单元	系统监测事件单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, 事件结束时间 date_time_s, 事件发生源 enum{系统 (0), 容器 (1), 应用 (2)}, 事件上报状态 array 通道上报状态, 告警类别 enum { CPU (0), 内存 (1), 存储 (2), 主板温度 (3) }, 实际值 long-unsigned (单位: %, 换算: -2), 设置阈值 long-unsigned (单位: %, 换算: -2), 容器名称 visible-string, 应用名称 visible-string, 服务名称 visible-string } 注: 如果没有容器名称、应用名称、服务名称则填写 NULL
3330	8	STA 入网事件单元	载波/无线模块入网事件单元::=structure { 事件记录序号 double-long-unsigned, 事件发生时间 date_time_s, }

OI	IC	对象名称	实例的对象属性及方法定义
			<p>事件结束时间 date_time_s,</p> <p>事件发生源 NULL,</p> <p>事件上报状态 array 通道上报状态,</p> <p>入网节点列表 array 入网节点信息</p> <p>}</p> <p>入网节点信息::=structure</p> <p>{</p> <p>从节点通信地址 octet-string,</p> <p>从节点协议类型 enum</p> <p>{</p> <p>未知 (0),</p> <p>DL/T 645-1997 (1),</p> <p>DL/T 645-2007 (2),</p> <p>DL/T 698.45 (3),</p> <p>CJ/T 188-2004 (4)</p> <p>},</p> <p>芯片管理 ID octet-string</p> <p>}</p>
3331	8	锁具异常事件记录单元	<p>锁具异常事件记录单元::=structure</p> <p>{</p> <p>事件记录序号 double-long-unsigned,</p> <p>事件发生时间 date_time_s,</p> <p>事件发生源 octet-string,</p> <p>事件上报状态 array 通道上报状态,</p> <p>锁具状态 bit-string,</p> <p>锁具状态变位 bit-string,</p> <p>第 1 个关联对象属性的数据 Data,</p> <p>...</p> <p>第 n 个关联对象属性的数据 Data</p> <p>}</p> <p>锁具状态:</p> <p>bit0-表示锁具开关状态, 置“0”: 锁关了, 置“1”: 锁开了</p> <p>bit1-表示锁具供电状态, 置“0”: 正常, 置“1”: 异常</p> <p>bit2-表示锁具供电窗状态, 置“0”: 正常, 置“1”: 异常</p> <p>bit3-表示锁具锁梁状态, 置“0”: 正常, 置“1”: 断裂</p> <p>bit4-表示锁具在线状态, 置“0”: 在线, 置“1”: 离线</p> <p>锁具状态变位:</p> <p>bit0~bit4 依次表示表示锁具开关状态变位、锁具供电状态变位、锁具供电窗状态变位、锁具锁梁状态变位、锁具在线状态变位, 置“0”: 未变位, 置“1”: 变位。</p> <p>事件发生源: 事件发生源为锁具编号。</p>

OI	IC	对象名称	实例的对象属性及方法定义
3401	7	充电计划生成失败事件	属性 2 (事件记录表) ::=array 充电计划生成失败记录单元 属性 6 (配置参数) ::=structure { }
3402	7	充电计划上报事件	属性 2 (事件记录表) ::=array 充电计划上报记录单元 属性 6 (配置参数) ::=structure { }
3403	7	充电申请事件	属性 2 (事件记录表) ::=array 充电申请事件记录单元 属性 6 (配置参数) ::=structure { }
3404	7	充电执行事件	属性 2 (事件记录表) ::=array 充电执行事件记录单元 属性 6 (配置参数) ::=structure { }
3405	7	充电桩异常事件	属性 2 (事件记录表) ::=array 有序充电类事件记录单元 属性 6 (配置参数) ::=structure { }
3406	7	充电订单结算事件	属性 2 (事件记录表) ::=array 充电订单结算记录单元 属性 6 (配置参数) ::=structure { }
3407	7	充电计划下发失败事件	属性 2 (事件记录表) ::=array 标准事件记录单元 属性 6 (配置参数) ::=structure { }
3500	7	监控单元遥信变位事件	属性 2 (事件记录表) ::=array 监控单元遥信变位事件记录单元 属性 6 (配置参数) ::=structure { } 事件发生源::=TSA 注: 事件发生源为监控单元地址。
3501	7	监控单元遥控记录事件	属性 2 (事件记录表) ::=array 监控单元遥控记录事件单元 属性 6 (配置参数) ::=structure { } 事件发生源::=TSA 注: 事件发生源为监控单元地址。
3600	7	系统监测事件	属性 2 (事件记录表) ::=array 系统监测事件单元 属性 6 (配置参数) ::=structure {

OI	IC	对象名称	实例的对象属性及方法定义
			内存使用率上限阈值 long-unsigned (单位: %, 换算: -2), 存储使用率上限阈值 long-unsigned (单位: %, 换算: -2), CPU 使用率上限阈值 long-unsigned (单位: %, 换算: -2), 主板温度上限阈值 long (单位: °C, 换算: -1), 监测周期 long-unsigned (单位: 分钟, 换算: 0) }
3601	7	容器监测事件	属性 2 (事件记录表) ::=array 系统监测事件单元 属性 6 (配置参数) ::=structure { 内存使用率上限阈值 long-unsigned (单位: %, 换算: -2), 存储使用率上限阈值 long-unsigned (单位: %, 换算: -2), CPU 使用率上限阈值 long-unsigned (单位: %, 换算: -2), 监测周期 long-unsigned (单位: 分钟, 换算: 0) }
3602	7	应用监测事件	属性 2 (事件记录表) ::=array 系统监测事件单元 属性 6 (配置参数) ::=structure { 内存使用率上限阈值 long-unsigned (单位: %, 换算: -2), CPU 使用率上限阈值 long-unsigned (单位: %, 换算: -2), 监测周期 long-unsigned (单位: 分钟, 换算: 0) }
3701	7	锁具异常事件	属性 2 (事件记录表) ::=array 锁具异常事件记录单元 属性 6 (配置参数) ::=structure { }

A.6 OIA1=4H

OIA1=4H的对象标识定义见表A.6。

表 A.6 OIA1=4H 对象标识定义

OI	IC	对象名称	实例的对象属性及方法定义
4000	8	日期时间	属性 2::=date_time_s 属性 3 (校时模式) ::=enum { 主站校时 (0), 心跳校时 (1), 北斗/GPS (2), 其它 (255) } 属性 4 (心跳校时参数) ::=structure { 最近心跳总个数 unsigned, 最大值剔除个数 unsigned, 最小值剔除个数 unsigned, 通讯延时阈值 unsigned (单位: 秒), 最少有效个数 unsigned } 属性 5 (广播校时参数) ::= structure { 最小校时偏差 long-unsigned (单位: 秒, 无换算), 最大校时偏差 long-unsigned (单位: 秒, 无换算) } 方法 127: 广播校时 (参数) 参数::=date_time_s 注: 最小校时偏差: 小于此范围所有校时均不响应, 包括明文广播校时、密文广播校时, 默认 60 秒; 最大校时偏差: 超过此范围的明文广播校时不响应, 密文广播校时响应, 默认 300 秒。
4001	8	通信地址	属性 2::=octet-string
4002	8	表号	属性 2::=octet-string
4003	8	客户编号	属性 2::=octet-string
4004	8	设备地理坐标	属性 2::=structure { 经度 structure {

OI	IC	对象名称	实例的对象属性及方法定义
			方位 enum{E (0), W (1) }, 度 unsigned, 分 unsigned, 秒 unsigned }, 纬度 structure { 方位 enum{S (0), N (1) }, 度 unsigned, 分 unsigned, 秒 unsigned }, 高度 (cm) double-long-unsigned }
4005	8	组地址	属性 2::=array octet-string
4006	8	时钟源	属性 2 (只读) ::=structure { 时钟源 enum { 处理器内部 (0), 时钟芯片 (1), 互联网时钟 (2), 卫星时钟 (3), 长波时钟 (4) }, 状态 enum{可用 (0), 不可用 (1) } } 方法 127: 启用 (参数) 参数::=NULL 方法 128: 禁用 (参数) 参数::=NULL
4007	8	LCD 参数	属性 2::=structure { 上电全显时长 unsigned, 背光点亮时长 long-unsigned (按键时背光点亮时长), 显示查看背光点亮时长 long-unsigned, 无电按键屏幕驻留最大时长 long-unsigned, 显示电能小数位数 unsigned, 显示功率 (最大需量) 小数位数 unsigned, 液晶①②字样意义 unsigned } 以上时长的单位均为: 秒。 液晶①②字样意义: 0 显示当前套、备用套时段, 1 显示当前套、备用套费率。

OI	IC	对象名称	实例的对象属性及方法定义
4008	8	备用套时区表切换时间	属性 2::=date_time_s 秒=FFH
4009	8	备用套日时段切换时间	属性 2::=date_time_s 秒=FFH
400A	8	备用套分时费率切换时间	属性 2::=date_time_s 秒=FFH
400B	8	备用套阶梯电价切换时间	属性 2::=date_time_s 秒=FFH
400C	8	时区时段数	属性 2::=structure { 年时区数(p≤14) unsigned, 日时段表数(q≤8) unsigned, 日时段数(每日切换数)(m≤14) unsigned, 费率数(k≤63) unsigned, 公共假日数(n≤254) unsigned }
400D	8	阶梯数	属性 2::=unsigned, 无单位, 无换算
400E	8	谐波分析次数	属性 2::=unsigned, 无单位, 无换算 次数表示从 2 到“次数”区间范围。
400F	8	密钥总条数	属性 2::=unsigned, 无单位, 无换算
4010	8	计量元件数	属性 2(只读)::=unsigned, 无单位, 无换算 计量元件数单相表为 1, 三相三线表为 2, 三相四线表为 3。
4011	8	公共假日表	属性 2::=array 公共假日 公共假日::=structure { 日期 date, 日时段表号 unsigned }
4012	8	周休日特征字	属性 2::=bit-string(SIZE(8)), 见附录 G.10。
4013	8	周休日采用的日时段表号	属性 2::=unsigned, 无单位, 无换算
4014	8	当前套时区表	属性 2::=array 时区 时区::=structure { 月 unsigned, 日 unsigned, 日时段表号 unsigned }
4015	8	备用套时区表	同 4014
4016	8	当前套日时段表	属性 2::=array 日时段表 日时段表::=array 时段 时段::=structure

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> { 时 unsigned, 分 unsigned, 费率号 unsigned } </pre> 费率号：该时段采用的费率号。
4017	8	备用套日时段表	同 4016
4018	8	当前套费率电价	属性 2（只读） ::=array 费率电价 费率电价 ::=double-long-unsigned 单位：元/kWh，换算：-4
4019	8	备用套费率电价	属性 2 ::=array 费率电价 费率电价 ::=double-long-unsigned 单位：元/kWh，换算：-4
401A	8	当前套阶梯电价	属性 2(阶梯参数，只读) ::=structure <pre> { 阶梯值数组 array 阶梯值, 阶梯电价数组 array 阶梯电价, 阶梯结算日数组 array 阶梯结算日 } </pre> 阶梯值 ::=double-long-unsigned 阶梯电价 ::=double-long-unsigned 阶梯结算日 ::=structure <pre> { 月 unsigned, 日 unsigned, 时 unsigned } </pre> 阶梯值：单位：kWh，换算：-2 阶梯电价：单位：元/kWh，换算：-4 阶梯结算日中月、日、时均有效时则按结算日执行阶梯冻结，如果第 1 阶梯结算日中仅日、时有效则每月该日时执行阶梯冻结，当全部无效时不执行阶梯冻结。
401B	8	备用套阶梯电价	同 401A
401C	8	电流互感器变比	属性 2 ::=double-long-unsigned，无单位，无换算
401D	8	电压互感器变比	属性 2 ::=double-long-unsigned，无单位，无换算
401E	8	报警金额限值	属性 2（参数） ::=structure <pre> { 报警金额限值 1 double-long-unsigned, 报警金额限值 2 double-long-unsigned, } </pre> 单位：元，换算：-2

OI	IC	对象名称	实例的对象属性及方法定义
401F	8	其它金额限值	属性 2 (参数) ::=structure { 透支金额限值 double-long-unsigned, 囤积金额限值 double-long-unsigned, 合闸允许金额限值 double-long-unsigned, } 单位: 元, 换算: -2
4020	8	报警电量限值	属性 2 ::=structure { 报警电量限值 1 double-long-unsigned, 报警电量限值 2 double-long-unsigned, } 单位: kWh, 换算: -2
4021	8	其它电量限值	属性 2 ::=structure { 囤积电量限值 double-long-unsigned, 透支电量限值 double-long-unsigned, 合闸允许电量限值 double-long-unsigned, } 单位: kWh, 换算: -2
4022	6	插卡状态字	属性 2 (只读) ::=bit-string(SIZE(16)) 定义见 G.11 。
4024	8	剔除	属性 2 ::=enum {剔除投入 (1), 剔除解除 (2)}
4025	8	采集器远程升级结果表	属性 2 (升级结果列表, 只读) ::=array 采集器升级结果 属性 3 (采集器升级控制参数) ::=structure { 允许升级广播轮次数 unsigned, 允许升级点对点补发天数 unsigned }
4026	8	采集器升级结果	属性 2 (升级结果) ::=structure { 序号 long-unsigned, 采集器地址 TSA, 采集器升级结果标识 unsigned, 补发开始时间 date_time_s, 升级成功时间 date_time_s, 广播成功块数 long-unsigned, 补发块数 long-unsigned, 升级前采集器版本 VersionInfo, 升级后采集器版本 VersionInfo }

OI	IC	对象名称	实例的对象属性及方法定义
			VersionInfo 见表 174 属性 3 定义。 采集器升级结果标识： 00：其他原因； 01：补发启动传输失败； 02：补发失败； 55：补发成功； AA：广播成功； EE：异常终止。
4027	8	升级结果	属性 2（升级结果，只读） ::=enum { 初始状态(出厂默认) (0)， 启动传输成功，待下载 (1)， 下载中 (2)， 下载成功 (3)， 真实性、完整性校验成功 (4)， 定时升级等待中 (5)， 升级成功 (6)， 硬件版本不匹配 (20)， 软件版本不匹配 (21)， 写外部 Flash 错 (22)， 读取外部 Flash 错 (23)， 写程序 Flash 错 (24)， 完整性校验错 (25)， 真实性校验错 (26)， 定时升级时间异常 (27)， 升级失败其它原因 (255) }
4028	8	逻辑设备序列号	属性 2（只读） ::=octet-string
4030	8	电压合格率参数	属性 2 ::=structure { 电压考核上限 long-unsigned（单位：V，换算：-1）， 电压考核下限 long-unsigned（单位：V，换算：-1）， 电压合格上限 long-unsigned（单位：V，换算：-1）， 电压合格下限 long-unsigned（单位：V，换算：-1） }
4031	8	电能表模组列表	属性 2（只读） ::= array 模组 模组 ::=structure { 模组描述符 visible-string， 逻辑地址 unsigned， 模组类别 bit-string， }

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> 模块序列号 octet-string } 属性 3（模块版本信息列表） ::= array 模块版本信息 模块版本信息 ::= structure { 模块逻辑地址 unsigned, 厂商代码 visible-string(SIZE(4)), 软件版本号 visible-string(SIZE(16)), 软件版本日期 visible-string(SIZE(6)), 硬件版本号 visible-string(SIZE(16)), 硬件版本日期 visible-string(SIZE(6)), 厂商扩展信息 visible-string(SIZE(16)) } 方法 127: APP 信息文件传送（参数） 参数 ::= octet-string 方法 128: APP 数据传送（参数） 参数 ::= array 数据传送 数据传送 ::= structure { 对象属性描述符 OAD, 操作前数据 Data OPTIONAL, 操作后数据 Data OPTIONAL } </pre>
4040	8	TA 专用模块	<pre> 属性 2 模块信息（只读） ::= structure { 模块 ID octet-string (SIZE (8)) , 厂商代号 visible-string(SIZE(4)), 设备编号 visible-string(SIZE (8)), 运行区软件版本号 visible-string(SIZE(4)), 运行区软件发布日期 date, 备份区软件版本号 visible-string(SIZE(4)), 备份区软件发布日期 date, 配置容量信息码 visible-string(SIZE(11)) , 通信协议版本号 visible-string(SIZE(4)) , 软件版本号 visible-string(SIZE (4)) , 硬件发布日期 date } 属性 3 安全认证信息（只读） ::= structure { 安全模块序列号 octet-string (SIZE (8)) , 当前密钥状态 octet-string (SIZE (4)) , 双向认证状态 octet-string (SIZE (4)) } </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> } 属性 4 电流回路状态（只读） ::= array enum 属性 5 频率最大值（只读） ::= array double-long-unsigned 属性 6 频率最小值（只读） ::= array double-long-unsigned 属性 7 工频电流有效值（只读） ::= array double-long-unsigned 属性 8 第一组阻抗模值（只读） ::= array double-long-unsigned 属性 9 第二组阻抗模值（只读） ::= array double-long-unsigned 属性 10 第三组阻抗模值（只读） ::= array double-long-unsigned 属性 11 第一组阻抗角度（只读） ::= array double-long-unsigned 属性 12 第二组阻抗角度（只读） ::= array double-long-unsigned 属性 13 第三组阻抗角度（只读） ::= array double-long-unsigned 属性 14 第一组阻抗频率（只读） ::= array double-long-unsigned 属性 15 第二组阻抗频率（只读） ::= array double-long-unsigned 属性 16 第三组阻抗频率（只读） ::= array double-long-unsigned 属性 17 换算及单位（只读） ::=structure { 电流有效值单位换算 Scaler_Unit（单位： A， 换算： -3）， 频率单位换算 Scaler_Unit（单位： Hz， 换算： 0）， 阻抗模值单位换算 Scaler_Unit（单位： Ω， 换算： -2）， 阻抗角度单位换算 Scaler_Unit（单位： °， 换算： -1）， 温度单位换算 Scaler_Unit（单位： °C， 换算： -1） } 方法 127 密钥更新（参数） 参数 ::=structure { 本帧密钥条数 unsigned, 密钥文件 octet-string } 应答 ::=bool（True: 成功, False: 失败） </pre>
4041	8	电流回路监测使能	<pre> 属性 2 ::= structure { A相使能 bool, B相使能 bool, C相使能 bool } </pre>
4047	8	当前计划时段信息	<pre> 属性 2 ::=structure { 当前计划时段起始时间 structure { 日 unsigned, 时 unsigned, </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			分 unsigned } 当前计划时段结束时间 structure { 日 unsigned, 时 unsigned, 分 unsigned } 当前计划时段购电量 long64-unsigned (单位: kWh, 换算: -4) } 属性 3 (当前计划时段实际用电量) ::=long64- unsigned 单位: kWh, 换算: -4
4100	8	最大需量周期	属性 2::=unsigned, 单位: 分钟, 换算: 0
4101	8	滑差时间	属性 2::=unsigned, 单位: 分钟, 换算: 0
4102	8	校表脉冲宽度	属性 2 (只读) ::=unsigned, 单位: 毫秒, 换算: 0
4103	8	资产管理码	属性 2::=visible-string(SIZE(32))
4104	8	额定电压	属性 2 (只读) ::=visible-string(SIZE(6))
4105	8	额定电流/基本电流	属性 2 (只读) ::=visible-string(SIZE(6))
4106	8	最大电流	属性 2 (只读) ::=visible-string(SIZE(6))
4107	8	有功准确度等级	属性 2 (只读) ::=visible-string(SIZE(4))
4108	8	无功准确度等级	属性 2 (只读) ::=visible-string(SIZE(4))
4109	8	电能表有功常数	属性 2 (只读) ::=double-long-unsigned 单位: imp/kWh, 换算: 0
410A	8	电能表无功常数	属性 2 (只读) ::=double-long-unsigned 单位: imp/kvarh, 换算: 0
410B	8	电能表型号	属性 2 (只读) ::=visible-string(SIZE(32))
410C	8	ABC 相电导系数	属性 2::=structure { A 相电导 long (单位: 无, 换算: -3), B 相电导 long (单位: 无, 换算: -3), C 相电导 long (单位: 无, 换算: -3) }
410D	8	ABC 相电抗系数	属性 2::=structure { A 相电抗 long (单位: 无, 换算: -3), B 相电抗 long (单位: 无, 换算: -3), C 相电抗 long (单位: 无, 换算: -3) }
410E	8	ABC 相电阻系数	属性 2::=structure { A 相电阻 long (单位: 无, 换算: -3),

OI	IC	对象名称	实例的对象属性及方法定义
			B相电阻 long (单位: 无, 换算: -3), C相电阻 long (单位: 无, 换算: -3) }
410F	8	ABC相电纳系数	属性2::=structure { A相电纳 long (单位: 无, 换算: -3), B相电纳 long (单位: 无, 换算: -3), C相电纳 long (单位: 无, 换算: -3) }
4111	8	软件备案号	属性2 (只读) ::=visible-string
4112	8	有功组合方式特征字	属性2::=bit-string(SIZE(8)), 见附录G。
4113	8	无功组合方式1特征字	属性2::=bit-string(SIZE(8)), 见附录G。
4114	8	无功组合方式2特征字	属性2::=bit-string(SIZE(8)), 见附录G。
4116	8	结算日	属性2 (配置参数) ::=array 结算日日期 结算日日期::=structure { 日 unsigned, 时 unsigned }
4117	8	期间需量冻结周期	属性2 (配置参数) ::=TI
4204	8	终端对电能表广播校时	属性2 (广播校时参数) ::=structure { 广播校时启动时间 time, 启用标志 bool } 属性3 (单地址广播校时参数) ::=structure { 时钟误差阈值 integer (单位: 秒), 终端广播校时启动时间 time, 启用标志 bool } 电能表时钟误差由终端计算获得。执行单地址广播校时后生成终端对电能表校时事件。
4300	19	电气设备	方法127 修改液晶设置密码 (参数) 参数::=visible-string(SIZE(6))
4307	19	水表	方法127: 出厂启用 (参数) 参数::=NULL 出厂前发出, 且只能发一次。 方法128: 阀门控制 (参数) 参数::=enum{开阀 (55H), 关阀 (99H)}。 方法129: 机电同步 (double-long-unsigned)

OI	IC	对象名称	实例的对象属性及方法定义
			用于出厂前机电同步，仅允许执行一次。
4308	19	气表	方法 127：出厂启用（参数） 参数： <code>::=NULL</code> 出厂前发出，且只能发一次。 方法 128：阀门控制（参数） 参数： <code>::=enum{开阀（55H），关阀（99H）}</code> 。 方法 129：机电同步（ <code>double-long-unsigned</code> ） 用于出厂前机电同步，仅允许执行一次。
4309	19	热表	方法 127：出厂启用（参数） 参数： <code>::=NULL</code> 出厂前发出，且只能发一次。 方法 128：阀门控制（参数） 参数： <code>::=enum{开阀（55H），关阀（99H）}</code> 。 方法 129：机电同步（参数） 参数： <code>::=structure</code> <pre> { 热量 double-long-unsigned, 热流量（水流量） double-long-unsigned } </pre> 用于出厂前机电同步，仅允许执行一次。
4310	8	智能锁具	属性 11（锁具开闭状态，只读）： <code>::=enum{闭（0），开（1）}</code> 属性 12（开锁码）： <code>::=octet-string</code> 属性 13（供电状态，只读）： <code>::=enum{正常（0），异常（1），缺省（2）}</code> 属性 14（供电窗状态，只读）： <code>::=enum{正常（0），异常（1），缺省（2）}</code> 属性 15（锁梁状态，只读）： <code>::=enum{正常（0），异常（1），缺省（2）}</code> 属性 16（SM4 密码）： <code>::=octet-string（16）</code> 属性 17（计数器同步标识）： <code>::=enum{计数器同步成功（0），计数器未同步（1）}</code> 属性 18（在线状态，只读）： <code>::=enum{在线（0），离线（1）}</code> 方法 127：开锁（参数） 参数： <code>::=octet-string</code> 方法 128：闭锁（参数） 参数： <code>::=octet-string</code>
4311	28	热水器	属性 5（运行信息）： <code>::=structure</code> <pre> { 当前温度 long（单位：℃，换算：-1）， 加热湿度 long（单位：℃，换算：-1）， 加热最高温度 long（单位：℃，换算：-1）， 延迟开机剩余时间 long-unsigned（单位：分钟，换算：0）， 延迟关机剩余时间 long-unsigned（单位：分钟，换算：0）， 设定功率 double-long（单位：W，换算：-1）， } </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			设定整机功率负荷百分比 long-unsigned (单位: %, 换算: -2), 故障码 long-unsigned }
4312	30	洗衣机	属性 5 (运行信息) ::=structure { 工作状态 unsigned, 工作模式 unsigned, 水位 unsigned, 设定温度 long (单位: °C, 换算: -1), 设定转数 long (单位: °C, 换算: -1), 设定洗涤时间 long-unsigned (单位: 秒, 换算: 0), 预约时间 long-unsigned (单位: 分钟, 换算: 0), 洗衣剩余时间 long-unsigned (单位: 秒, 换算: 0), 设定功率 double-long (单位: W, 换算: -1), 设定整机功率负荷百分比 long-unsigned (单位: %, 换算: -2), 故障码 long-unsigned }
4313	28	智能插座	
4314	28	空调	属性 5 (运行信息) ::=structure { 设定温度 long (单位: °C, 换算: -1), 设定湿度 long-unsigned (单位: %, 换算: -2), 工作模式 enum { 自动 (1), 制冷 (2), 除湿 (3), 制热 (4), 送风 (5), 自动除湿 (6), 手动除湿 (7) }, 设定风速 unsigned, 室内温度 long (单位: °C, 换算: -1), 室外温度 long (单位: °C, 换算: -1), 延迟开机剩余时间 long-unsigned (单位: 分钟, 换算: 0), 延迟关机剩余时间 long-unsigned (单位: 分钟, 换算: 0), 设定功率 double-long (单位: W, 换算: -1), 设定整机功率负荷百分比 long-unsigned (单位: %, 换算: -2), 故障码 long-unsigned }

OI	IC	对象名称	实例的对象属性及方法定义
4315	8	锁具状态监测器	属性 2（锁具设备列表） ::=array 锁具 锁具 ::=structure { 锁具编号 octet-string } 方法 127：给锁具状态监测器添加一个档案（参数） 参数 ::=octet-string（锁具编号） 方法 128：删除一个档案（参数） 参数 ::=octet-string（锁具编号） 方法 129：计数器同步（参数） 参数 ::=NULL 方法 130：蓝牙广播信息收悉（参数） 参数 ::=NULL
4400	20	应用连接	
4401	8	认证密码	属性 2（只写） ::=visible-string
4500	25	无线远程通信模块 1	
4501	25	无线远程通信模块 2	
4510	26	以太网通信模块 1	
4511	26	以太网通信模块 2	
4512	26	以太网通信模块 3	
4513	26	以太网通信模块 4	
4514	26	以太网通信模块 5	
4515	26	以太网通信模块 6	
4516	26	以太网通信模块 7	
4517	26	以太网通信模块 8	
4600	8	充电控制	属性 2（充电机参数，只读） ::=structure { 最高输出电压 long-unsigned（单位：V，换算：-1）， 最低输出电压 long-unsigned（单位：V，换算：-1）， 最大输出电流 double-long（单位：A 换算：-3）， 最小输出电流 double-long（单位：A 换算：-3） } 属性 3（支持的 BMS 通信协议） ::=enum { GB/T 27930-2011（1）， GB/T 27930-2015（2） } 属性 4（遥信状态） ::=array bit-string 属性 5（充电启动完成状态） ::=array 充电启动完成单元 充电启动完成单元 ::=structure {

OI	IC	对象名称	实例的对象属性及方法定义
			<p>成功标志 unsigned, 失败原因 octet-string, }</p> <p>属性 6 (充电停止完成状态) ::=array 充电停止完成单元 充电停止完成单元 ::=structure { 成功标志 unsigned, 停止原因 octet-string, }</p> <p>属性 7 (计费控制单元错误信息) ::=array bit-string 属性 8 (充电控制器错误信息) ::=array bit-string 注: 属性 5~属性 8, 数组 0 为充电桩自身信息, 其它为充电枪相关信息。 方法 127: 启动充电 (参数) 参数 ::= structure { 充电接口标识 unsigned, 负荷控制开关 enum{启动 (1), 关闭 (2)} }</p> <p>返回结果 ::=structure { 充电接口标识 unsigned, 失败原因 unsigned }</p> <p>方法 128: 停止充电 (参数) 参数 ::= structure { 充电接口标识 unsigned, 停止原因 enum { 正常停止 (1), 计费控制单元故障停止 (2), 充电控制器故障停止 (3) } }</p> <p>返回结果 ::=structure { 充电接口标识 unsigned, 失败原因 unsigned }</p> <p>方法 129: 启动充电服务 (参数) 参数 ::= structure</p>

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> { 充电接口标识 unsigned } 返回结果::=structure { 充电接口标识 unsigned, 失败原因 unsigned } 方法 130: 停止充电服务 (参数) 参数::= structure { 充电接口标识 unsigned } 返回结果::=structure { 充电接口标识 unsigned, 失败原因 unsigned } 方法 131: 电子锁上锁 (参数) 参数::= structure { 充电接口标识 unsigned, 电子锁序号 unsigned } 返回结果::=structure { 充电接口标识 unsigned, 电子锁序号 unsigned, 失败原因 unsigned } 方法 132: 电子锁解锁 (参数) 参数::= structure { 充电接口标识 unsigned, 电子锁序号 unsigned } 返回结果::=structure { 充电接口标识 unsigned, 电子锁序号 unsigned, 失败原因 unsigned </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> } 方法 133: 绝对值功率调节 (参数) 参数 ::= structure { 充电接口标识 unsigned, 调节参数 double-long (单位: kW, 换算: -4) } 返回结果 ::= structure { 充电接口标识 unsigned, 失败原因 unsigned } 方法 134: 百分比功率调节 (参数) 参数 ::= structure { 充电接口标识 unsigned, 调节参数 long-unsigned (单位: %, 换算: -1) } 返回结果 ::= structure { 充电接口标识 unsigned, 失败原因 unsigned } </pre>
4700	8	台区负荷预测及控制	<pre> 属性 2 (台区参数配置参数) ::= structure { 台区配变额定容量 double-long-unsigned (单位: kVA, 换算: -4), 最大负荷 double-long-unsigned (单位: kW, 换算: -4), 最大持续时间 double-long-unsigned (单位: S), 电压互感器变比 double-long-unsigned, 电流互感器变比 double-long-unsigned, 台区容量越限系数 long-unsigned(单位:%, 换算: -2), 台区启调系数 long-unsigned(单位:%, 换算: -2), 台区经济运行区间 array long-unsigned(单位:%, 换算: -2 分别表示 下限, 上限) , 正常充电台区容量配额系数 long-unsigned(单位:%, 换算: -2), 台区尖峰平谷时段电价 array double-long-unsigned (单位: 元/kW, 换算: -4) } 属性 3 台区基础负荷预测曲线 ::= array 时段负荷单元 属性 4 台区控制目标负荷曲线 ::= array 时段负荷单元 属性 5 临时台区基础负荷预测曲线 ::= array 时段负荷单元 </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			属性 6 临时台区控制目标负荷曲线 ::= array 时段负荷单元 方法 127: 添加或更新每日台区基础负荷预测曲线 (时段负荷单元) 方法 128: 更新临时台区基础负荷预测曲线 (时段负荷单元) 方法 129: 更新日台区控制目标负荷曲线 (时段负荷单元) 方法 130: 更新临时台区控制目标负荷曲线 (时段负荷单元)
4701	8	订单状态	属性 2 (订单状态, 只读) ::= structure <pre> { 充电申请单号 octet-string (SIZE(16)), 用户 ID visible-string, 路由器资产编号 visible-string (SIZE(22)), 执行状态 enum{正常执行 (1), 执行结束 (2), 执行失败 (3)}, 电能示值底值 array double-long, 当前电能示值 array double-long, 已充电量 array double-long, 已充时间 double-long-unsigned (单位: 秒), 计划充电功率 double-long (单位: kW, 换算: -4), 当前充电功率 double-long (单位: kW, 换算: -4), 电压 array long-unsigned, (单位: V, 换算: -1), 电流 array double-long, (单位: A 换算: -3), 充电桩状态 enum{待机 (1)、工作 (2)、故障 (3)} } </pre> 备注: 如果路由器未处于工作状态, 充电申请单号为全 0。
4702	8	充电服务	属性 2 (充电服务状态, 只读) ::=enum <pre> { 充电服务可用 (0), 充电服务停止 (1) } </pre> 属性 3 (充电通信状态, 只读) ::=enum <pre> { 正常 (0), 异常 (1) } </pre> 注释: 指与充电桩通信状态。 属性 4 (充电桩状态, 只读) ::= unsigned bit3~bit0 表示工作状态, 1: 待机, 2 工作, 3 故障, 4 充满; bit4 表示连接确认开关状态: 0 连接, 1 未连接; bit5 表示急停动作告警, 0 正常, 1 异常; bit6 表示避雷器故障, 0 正常, 1 故障。bit7 表示充电枪未归位, 0 正常, 1 异常。 <pre> } </pre> 方法 127: 启动 (参数)

OI	IC	对象名称	实例的对象属性及方法定义
			参数:: <code>NULL</code> 方法 128: 停止 (参数) 参数:: <code>NULL</code> 方法 129: 功率调节 (参数) 参数:: <code>structure</code> <pre>{ 有功功率设定 double-long (单位: kW, 换算: -4) }</pre> 注释: 路由器具有的对象
4703	8	路由器信息	属性 2:: <code>structure</code> <pre>{ 充电桩编号 visible-string (SIZE(16)), 充电接口标识 unsigned, 充电桩的安装地址 visible-string, 充电桩最低充电功率 double-long (单位: W, 换算: -1), 额定功率 double-long (单位: W, 换算: -1), 是否支持唤醒 enum{ 不支持 (0), 支持 (1) } }</pre>
4704	8	路由器白名单	属性 2:: <code>array structure</code> <pre>{ 用户 ID visible-string, 用户 token visible-string, 账户状态 enum{ 正常 (0), 欠费 (2) } }</pre>
4705	8	路由器密钥信息表	属性 2:: <code>array 路由器密钥信息单元</code> 方法 127: Add (array 路由器密钥信息单元) 添加一组路由器密钥信息单元 方法 128: Clear (参数) 参数:: <code>NULL</code> 清空路由器密钥信息表
4706	8	路由器密钥信息单元	属性 2:: <code>structure</code> <pre>{ 通信地址 TSA, 表号 octet-string, 密钥信息 octet-string, 密钥信息 MAC octet-string }</pre>
4707	8	时段负荷单元	属性 2:: <code>structure</code> <pre>{ 日期 date, 功率密度 TI, }</pre>

OI	IC	对象名称	实例的对象属性及方法定义
			时段起始时间 time, 功率设定值 array double-long (单位: kW, 换算: -4) } 备注: 功率密度指业务应用时功率调节的颗粒度。
4708	8	充电计划单单元	属性 2 (充电计划单单元, 只读) ::=structure { 充电申请单 octet-string, 用户 ID visible-string, 决策单元 enum { 主站 (2), 控制器 (2) }, 决策类型 enum { 生成 (2), 调整 (2) }, 决策时间 date_time_s, 路由器资产编号 visible-string (SIZE(22)), 充电需求电量 double-long-unsigned (单位: kWh, 换算: -2), 充电额定功率 double-long (单位: kW, 换算: -4), 充电模式 enum{正常 (0), 有序 (1) }, 充电时段 array 时段充电功率 } 时段充电功率 ::=structure { 开始时间 date_time_s, 结束时间 date_time_s, 充电功率 double-long (单位: kW, 换算: -4) }
4709	8	充电申请单单元	属性 2 (充电申请单单元, 只读) ::=structure { 充电申请单号 octet-string, 用户 ID visible-string, 路由器资产编号 visible-string (SIZE(22)), 充电需求电量 double-long-unsigned (单位: kWh, 换算: -2), 计划用车时间 date_time_s, 充电方式 enum{正常 (0), 有序 (1) }, 身份认证 Token visible-string (SIZE(32)) }
470A	8	邀约对象单元	属性 2 (邀约对象单元) ::=structure { 家电通信地址 octet-string, 执行开始时间 datetime_s, 执行结束时间 datetime_s, 调节参数列表 array 调节参数, 调节方式 enum{自动调节 (0), 手动调节 (1) } }

OI	IC	对象名称	实例的对象属性及方法定义
			调节参数 ::= structure { 对象属性描述 OAD, 属性值 Data }
4800	8	非介入式负荷感知模块	属性 2 (电器设备配置) ::= array 电器设备 电器设备 ::= structure { 设备类号 long-unsigned, 设备特征 Data } 属性 3 (电器设备当前数据集, 只读) ::= array 电器设备当前数据单元 属性 4 (电器设备冻结数据集, 只读) ::= array 电器设备冻结数据单元 属性 5 (电器设备采样数据集, 只读) ::= array 电器设备采样数据单元 方法 127 添加电器设备 (参数) 参数 ::= 电器设备 方法 128 删除电器设备 (设备类号) 设备类号 ::= long-unsigned 方法 129 更新电器设备特征 (参数) 参数 ::= 电器设备
4801	8	电器设备当前数据单元	属性 2 (电器设备当前数据单元) ::= structure { 设备类号 long-unsigned, 电器设备状态 enum {关机 (0), 运行 (1)}, 电器设备有功功率 double-long (单位: W, 换算-1), 电器设备正向有功电能 double-long-unsigned (单位: kWh, 换算-4) }
4802	8	电器设备冻结数据单元	属性 2 (电器设备冻结数据单元) ::= structure { 设备类号 long-unsigned, 冻结周期内有用电量 double-long-unsigned (单位: kWh, 换算-4) }
4803	8	电器设备采样数据单元	属性 2 (电器设备采样数据单元) ::= structure { 相位信息 enum {A 相 (1), B 相 (2), C 相 (3)}, 电压采样数据 double-long, 电流采样数据 double-long }
4804	8	计量配置信息	属性 2 (配置参数) ::= structure { 电压系数 array double-long-unsigned, }

OI	IC	对象名称	实例的对象属性及方法定义
			电流系数 array double-long-unsigned, 相位系数 array double-long-unsigned, 零线电流系数 double-long-unsigned, 偏移小数位数 unsigned, 采样周波点数 unsigned long, 采样数据长度 unsigned } 方法 127 启动采样数据输出 (参数) 参数::=NULL 方法 128 停止采样数据输出 (参数) 参数::=NULL
4900	27	监控单元集合对象	
4901	8	死区参数	属性 2::=structure { 电流死区 double-long (单位: A, 换算: -3), 交流电压死区 double-long-unsigned (单位: V, 换算: -2), 直流电压死区 double-long-unsigned (单位: V, 换算: -2), 功率死区 double-long (单位: W, 换算: -3), 频率死区 long (单位: 无, 换算: -3) }
4902	8	PT 参数	属性 2::=structure { 一次额定 long-unsigned (单位: kV, 换算: -1), 二次额定 long-unsigned (单位: kV, 换算: -1) }
4903	8	CT 参数	属性 2::=structure { 相 CT 一次额定 double-long (单位: A, 换算: -3), 相 CT 二次额定 double-long (单位: A, 换算: -3), 零序 CT 一次额定 double-long (单位: A, 换算: -3), 零序 CT 二次额定 double-long (单位: A, 换算: -3) }
4904	8	低电压报警	属性 2::=structure { 报警门限值 double-long-unsigned (单位: V, 换算: -2), 报警延时 long-unsigned (单位: 秒, 换算: 无) }
4905	8	过电压报警	属性 2::=structure { 报警门限值 double-long-unsigned (单位: V, 换算: -2), 报警延时 long-unsigned (单位: 秒, 换算: 无) }

OI	IC	对象名称	实例的对象属性及方法定义
			}
4906	8	重载报警	属性 2::=structure { 报警门限值 double-long (单位: A, 换算: -3), 报警延时 long-unsigned (单位: 秒, 换算: 无) }
4907	8	过载报警	属性 2::=structure { 报警门限值 double-long (单位: A, 换算: -3), 报警延时 long-unsigned (单位: 秒, 换算: 无) }
4908	8	开入量采集防抖时间	属性 2::=double-long-unsigned (单位: 秒, 换算: -2)
4909	8	分闸输出脉冲保持时间	属性 2::=structure { 分闸输出脉冲保持时间 double-long-unsigned (单位: 秒, 换算: -2), 合闸输出脉冲保持时间 double-long-unsigned (单位: 秒, 换算: -2) }
490A	8	蓄电池自动活化周期	属性 2::=structure { 蓄电池自动活化周期 double-long-unsigned (单位: 天, 换算: 无), 蓄电池自动活化时刻 double-long-unsigned (单位: 时, 换算: 无) }
490B	8	电池电压低报警阈值	属性 2::=long-unsigned (单位: V, 换算: -1)
490C	8	短路故障报警	属性 2::=structure { 突变电流阈值 double-long (单位: A, 换算: -3), 持续时间 double-long-unsigned (单位: 秒, 换算: -3) }
490E	8	录波电场下降比例	属性 2::=long-unsigned (单位: 无, 换算: -2)
490F	8	永久故障	属性 2::=enum { 上电复位 (0), 定时复位 (1) } 属性 3::=structure { 上电复归时间 double-long-unsigned (单位: 秒, 换算: -1), 确认时间 double-long-unsigned (单位: 秒, 换算: -1) }
4910	8	负荷电流超限阈值	属性 2::=double-long (单位: A, 换算: -3)
4911	8	过电流报警阈值	属性 2::=double-long (单位: A, 换算: -3)
4912	8	相线路有电阈值	属性 2::=long-unsigned (单位: V, 换算: -1)
4913	8	故障指示灯自动复归	属性 2::=structure { 是否投入 bool, }

OI	IC	对象名称	实例的对象属性及方法定义
			复归时间 double-long-unsigned (单位: 秒, 换算: -1) }
4914	8	故障遥信保持时间	属性 2::=double-long-unsigned (单位: 秒, 换算: -1)
4915	8	过流停电跳闸投退	属性 2::=bool
4916	8	过流参数	属性 2 (过流 I 段) ::=过流定值 属性 3 (过流 II 段) ::=过流定值 属性 4 (过流 III 段) ::=过流定值 属性 5 (零序过流) ::=过流定值 过流定值 ::=structure { 告警投入 bool, 出口投入 bool, 定值 double-long (单位: A, 换算: -3), 时间 double-long-unsigned (单位: 秒, 换算: -1) }
4917	8	重合闸	属性 2::=structure { 重合闸投入 bool, 大电流闭锁重合投入 bool, 重合闸次数 unsigned, 第一次重合闸时间 double-long-unsigned (单位: 秒, 换算: -1), 第二次重合闸时间 double-long-unsigned (单位: 秒, 换算: -1), 第三次重合闸时间 double-long-unsigned (单位: 秒, 换算: -1), 大电流闭重定值 double-long (单位: A, 换算: -3) }
4918	8	遮断电流闭锁	属性 2::=structure { 是否投入 bool, 定值 double-long (单位: A, 换算: -3) }
4919	8	就地重合器模式参数	属性 2::=structure { X 时间定值 double-long-unsigned (单位: 秒, 换算: -1), Y 时间定值 double-long-unsigned (单位: 秒, 换算: -1), S 时间定值 double-long-unsigned (单位: 秒, 换算: -1), 单相接地选线投入 bool, 接地选线跳闸时间 double-long-unsigned (单位: 秒, 换算: -1), 接地选线重合闸时间 double-long-unsigned (单位: 秒, 换算: -1), 自适应相间短路故障处理投入 bool, 自适应单相接地故障处理投入 bool, 接地故障确认时间定值 double-long-unsigned (单位: 秒, 换算: -1) }

OI	IC	对象名称	实例的对象属性及方法定义
			}

A.7 OIA1=5H

OIA1=5H的对象标识定义见表A.7。

表 A.7 OIA1=5H 对象标识定义

OI	IC	对象名称	实例的对象属性及方法定义
5000	9	瞬时冻结	
5001	9	秒冻结	
5002	9	分钟冻结	属性 5（深度存储关联对象属性表） ::=array 一个关联的对象属性 一个关联的对象属性 ::=structure { 冻结周期 long-unsigned, 关联对象属性描述符 OAD, 存储深度 double-long-unsigned } 方法 9：添加一个冻结对象属性（冻结对象） 冻结对象 ::=structure { 冻结周期 long-unsigned, 关联对象属性描述符 OAD, 存储深度 double-long-unsigned } 方法 10：批量添加冻结对象属性（array 冻结对象）
5003	9	小时冻结	
5004	9	日冻结	
5005	9	结算日冻结	
5006	9	月冻结	
5007	9	年冻结	
5008	9	时区表切换冻结	
5009	9	日时段表切换冻结	
500A	9	费率电价切换冻结	
500B	9	阶梯切换冻结	
5011	9	阶梯结算冻结	

A.8 0IA1=6H

0IA1=6H的对象标识定义见表A.8。

表 A.8 0IA1=6H 对象标识定义

OI	IC	对象名称	实例的对象属性及方法定义
6000	11	采集档案配置表	<p>属性 2（配置表，只读） ::=array 采集档案配置单元</p> <p>方法 127: Add（采集档案配置单元）</p> <p>方法 128: AddBatch（array 采集档案配置单元）</p> <p>方法 129: Update（参数）</p> <p>参数 ::=structure</p> <pre>{ 配置序号 long-unsigned, 基本信息 Basic_object }</pre> <p>方法 130: Update（参数）</p> <p>参数 ::=structure</p> <pre>{ 配置序号 long-unsigned, 扩展信息 Extended_object, 附属信息 Annex_object }</pre> <p>方法 131: Delete（配置序号） 通过配置序号删除配置单元。</p> <p>方法 132: Delete（基本信息） 通过基本信息对象删除配置单元。</p> <p>方法 133: Delete（参数）</p> <p>参数 ::=structure</p> <pre>{ 通信地址 TSA, 端口号 OAD }</pre> <p>通过通信地址及端口删除配置单元。</p> <p>方法 134: Clear（参数）</p> <p>参数 ::=NULL</p> <p>清空采集档案配置表。</p> <p>方法 135: Update（参数）</p> <p>参数 ::=structure</p> <pre>{ 通信地址 TSA, 附属信息 Annex_object }</pre>

OI	IC	对象名称	实例的对象属性及方法定义
			} 说明：增加更新档案附属信息方法
6001	8	采集档案配置单元	<pre> 属性 2 (Acquisition document definition) ::=structure { 配置序号 long-unsigned, 基本信息 Basic_object, 扩展信息 Extended_object, 附属信息 Annex_object } Basic_object ::=structure { 通信地址 TSA, 波特率 enum { 300bps (0), 600bps (1), 1200bps (2), 2400bps (3), 4800bps (4), 7200bps (5), 9600bps (6), 19200bps (7), 38400bps (8), 57600bps (9), 115200bps (10), 自适应 (255) }, 规约类型 enum { 未知 (0), DL/T 645-1997 (1), DL/T 645—2007 (2), DL/T 698.45 (3), CJ/T 188—2004 (4) }, 端口 OAD, 通信密码 octet-string, 费率个数 unsigned, 用户类型 unsigned, 接线方式 enum { 未知 (0), 单相 (1), 三相三线 (2), 三相四线 (3) }, 额定电压 long-unsigned(换算-1, 单位 V), 额定电流 long-unsigned(换算-1, 单位 A) } </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> Extended_object ::= structure { 采集器地址 TSA, 资产号 octet-string, 电压互感器变比 long-unsigned, 电流互感器变比 long-unsigned } Annex_object ::= array structure { 对象属性描述 OAD, 属性值 Data } </pre>
6002	11	搜表	<pre> 属性 2（所有搜表结果） ::= array 一个搜表结果 属性 5（跨台区搜表结果） ::= array 一个跨台区结果 属性 6（所有搜表结果记录数） ::= long-unsigned 属性 7（跨台区搜表结果记录数） ::= long-unsigned 属性 8 ::= structure { 每天周期搜表启用标志 bool, 自动更新采集档案 bool, 搜表事件生成标志 bool, 清空搜表结果选项 enum { 不清空 (0), 每天周期搜表前清空 (1), 每次搜表前清空 (2) } } 属性 9（每天周期搜表参数配置） ::= array 定时搜表参数 定时搜表参数 ::= structure { 开始时间 time, 搜表时长 (min) long-unsigned } 属性 10（搜表状态） ::= enum { 空闲 (0), 搜表中 (1) } 方法 127：启动搜表（搜表时长） 搜表时长 ::= long-unsigned, 单位：分钟，表示搜表持续时间，0 表示不限时间直至搜表结束。 </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			方法 128: 清空搜表结果(参数) 参数::=NULL 方法 129: 清空跨台区搜表结果(参数) 参数::=NULL
6003	8	一个搜表结果	一个搜表结果::=structure { 通信地址 TSA, 所属采集器地址 TSA, 规约类型 enum { 未知 (0), DL/T 645—1997 (1), DL/T 645—2007 (2), DL/T 698.45 (3), CJ/T 188—2004 (4) }, 相位 enum{未知 (0), A (1), B (2), C (3)}, 信号品质 unsigned, 搜到的时间 date_time_s, 搜到的附加信息 array 附加信息 } 附加信息::=structure { 对象属性描述 OAD, 属性值 Data }
6004	8	一个跨台区结果	一个跨台区结果::=structure { 通信地址 TSA, 主节点地址 TSA, 变更时间 date_time_s }
6005	8	户表新增标识	属性 2 ::=enum{非新增用户 (0), 新增用户 (1)}
6012	10	任务配置表	属性 2 (配置表) ::=array 任务配置单元 属性 3 (记录表) ::=array 记录单元 记录单元::=structure { 采集启动时标 date_time_s, 采集成功时标 date_time_s, 采集存储时标 date_time_s, 采集通信地址 TSA, }

OI	IC	对象名称	实例的对象属性及方法定义
			采集的数据 1 Data, ... 采集的数据 N Data } 方法 127: Add (array 任务配置单元) 添加或更新一组任务配置单元。 方法 128: Delete (array 任务 ID) 删除一组配置单元。 方法 129: Clear (参数) 参数::=NULL 清空任务配置表。 方法 130: Update (参数) 参数::=structure { 任务 ID unsigned, 状态 enum } 更新任务状态。
6013	8	任务配置单元	属性 2 (任务配置单元) ::=structure { 任务 ID unsigned, 执行时间间隔 TI, 方案类型 enum { 普通采集方案 (1), 事件采集方案 (2), 透明方案 (3), 上报方案 (4), 脚本方案 (5) }, 方案编号 unsigned, 开始时间 date_time_s, 结束时间 date_time_s, 延时 TI, 执行优先级 unsigned, 状态 enum{正常 (1), 停用 (2)}, 任务开始前脚本 id long-unsigned, 任务完成后脚本 id long-unsigned, 任务运行时段 structure } 当方案类型为脚本时, 方案编号为脚本 id。 任务运行时段 ::=structrue {

OI	IC	对象名称	实例的对象属性及方法定义
			类型 enum { 前闭后开 (0), 前开后闭 (1), 前闭后闭 (2), 前开后开 (3) }, 时段表 array 时段 } 时段::=structure { 起始小时 unsigned, 起始分钟 unsigned, 结束小时 unsigned, 结束分钟 unsigned }
6014	11	普通采集方案集	属性 2::=array 普通采集方案 方法 127: Add (array 普通采集方案) 添加或更新一组普通采集方案。 方法 128: Delete (array 方案编号) 删除一组普通采集方案。 方法 129: Clear (参数) 参数::=NULL 清空普通采集方案集。 方法 130: Set_CSD (参数) 参数::=structure { 方案编号 unsigned, 记录列选择 array CSD } 重置方案的记录列选择。
6015	8	普通采集方案	属性 2 (普通采集方案)::=structure { 方案编号 unsigned, 存储深度 long-unsigned, 采集方式 structure { 采集类型 unsigned, 采集内容 Data }, 记录列选择 array CSD,

OI	IC	对象名称	实例的对象属性及方法定义																		
			<p>表计集合 MS,</p> <p>存储时标选择 enum</p> <pre>{ 未定义 (0), 任务开始时间 (1), 相对当日 0 点 0 分 (2), 相对上日 23 点 59 分 (3), 相对上日 0 点 0 分 (4), 相对当月 1 日 0 点 0 分 (5), 数据冻结时标 (6), 相对上月月末 23 点 59 分 (7) }</pre> <p>采集方式数据格式如下:</p> <table border="1" data-bbox="754 880 1442 1137"> <thead> <tr> <th>采集类型</th> <th>采集内容</th> <th>表示</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>NULL</td> <td>采集当前数据</td> </tr> <tr> <td>1</td> <td>unsigned</td> <td>采集上第 N 次</td> </tr> <tr> <td>2</td> <td>NULL</td> <td>按冻结时标采集</td> </tr> <tr> <td>3</td> <td>TI</td> <td>按时标间隔采集</td> </tr> <tr> <td>4</td> <td>RetryMetering</td> <td>补抄</td> </tr> </tbody> </table> <pre>RetryMetering ::= structure { 数据时标间隔 TI, 补抄周期 (上 N 个) long-unsigned }</pre>	采集类型	采集内容	表示	0	NULL	采集当前数据	1	unsigned	采集上第 N 次	2	NULL	按冻结时标采集	3	TI	按时标间隔采集	4	RetryMetering	补抄
采集类型	采集内容	表示																			
0	NULL	采集当前数据																			
1	unsigned	采集上第 N 次																			
2	NULL	按冻结时标采集																			
3	TI	按时标间隔采集																			
4	RetryMetering	补抄																			
6016	11	事件采集方案集	<p>属性 2::=array 事件采集方案</p> <p>属性 5::=array 事件采集方案上报通道</p> <p>事件采集方案上报通道::=structure</p> <pre>{ 方案编号 unsigned, 上报通道 array OAD }</pre> <p>方法 127: Add (array 事件采集方案) 添加或更新一组事件采集方案。</p> <p>方法 128: Delete (array 方案编号) 删除一组事件采集方案。</p> <p>方法 129: Clear (参数) 参数::=NULL 清空事件采集方案集。</p> <p>方法 130: UpdateReportFlag (参数)</p> <p>方法 131: UpdateReportChannel (参数)</p>																		

OI	IC	对象名称	实例的对象属性及方法定义															
			参数 ::= structure { 方案编号 unsigned, 上报通道 array OAD } 参数 ::= structure { 方案编号 unsigned, 上报标识 bool }															
6017	8	事件采集方案	属性 2 (事件采集方案 Event acq plan) ::= structure { 方案编号 unsigned, 采集方式 structure { 采集类型 unsigned, 采集内容 Data } 表计集合 MS, 上报标识 bool (True: 立即上报, False: 不上报), 存储深度 long-unsigned } 采集方式数据格式如下: <table border="1" data-bbox="756 1256 1442 1469"> <thead> <tr> <th>采集类型</th> <th>采集内容</th> <th>表示</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>array ROAD</td> <td>周期采集事件数据</td> </tr> <tr> <td>1</td> <td>NULL</td> <td>根据通知采集所有事件数据</td> </tr> <tr> <td>2</td> <td>array ROAD</td> <td>根据通知采集指定事件数据</td> </tr> <tr> <td>3</td> <td>NULL</td> <td>根据通知存储生成的事件数据</td> </tr> </tbody> </table>	采集类型	采集内容	表示	0	array ROAD	周期采集事件数据	1	NULL	根据通知采集所有事件数据	2	array ROAD	根据通知采集指定事件数据	3	NULL	根据通知存储生成的事件数据
采集类型	采集内容	表示																
0	array ROAD	周期采集事件数据																
1	NULL	根据通知采集所有事件数据																
2	array ROAD	根据通知采集指定事件数据																
3	NULL	根据通知存储生成的事件数据																
6018	11	透明方案集	属性 2 ::= array 透明方案 方法 127: Add (透明方案) 添加更新一个透明方案或添加一组方案内容。 方法 128: AddMeterFrame (参数) 参数 ::= structure { 方案编号 long-unsigned, 通信地址 TSA, 方案控制标志 structure, 方案报文集 array 方案报文 } 添加一组报文。 方法 129: Delete (参数)															

OI	IC	对象名称	实例的对象属性及方法定义
			<p>参数 ::= structure</p> <pre>{ 方案编号 unsigned, 通信地址集合 array TSA }</pre> <p>删除一个方案的一组方案内容。</p> <p>方法 130: Delete (array 方案编号)</p> <p>删除一组透明方案。</p> <p>方法 131: Clear (参数)</p> <p>参数 ::= NULL</p> <p>清空透明方案集。</p>
6019	8	透明方案	<p>属性 2 (透明方案) ::= structure</p> <pre>{ 方案编号 unsigned, 方案内容集 array 方案内容, 存储深度 long-unsigned }</pre> <p>方案内容 ::= structure</p> <pre>{ 序号 long-unsigned, 通信地址 TSA, 开始前脚本 id long-unsigned, 完成后脚本 id long-unsigned, 方案控制标志 structure, 方案报文集 array 方案报文 }</pre> <p>方案控制标志 ::= structure</p> <pre>{ 上报透明方案结果并等待后续报文 bool, 等待后续报文超时时间 (秒) long-unsigned, 结果比对标识 enum { 不比对 (0), 比对 (1), 比对上报 (2) }, 结果比对参数 structure }</pre> <p>方案报文 ::= structure</p> <pre>{ 报文序号 unsigned,</pre>

OI	IC	对象名称	实例的对象属性及方法定义
			报文内容 octet-string } 结果比对参数 ::= structure { 特征字节 unsigned, 截取开始 long-unsigned, 截取长度 long-unsigned }
601A	11	透明方案结果集	属性 2 ::= array 一个透明方案结果
601B	8	一个透明方案结果	一个透明方案结果 ::= structure { 方案编号 unsigned, 方案执行时间 date_time_s, 通信地址 TSA, 结果集 array 结果 } 结果 ::= structure { 报文序号 unsigned, 报文响应时间 date_time_s, 命令结果 octet-string }
601C	11	上报方案集	属性 2 ::= array 上报方案 方法 127: Add (array 上报方案) 添加或更新一组上报方案。 方法 128: Delete (array 方案编号) 删除一组上报方案。 方法 129: Clear (参数) 参数 ::= NULL 清空上报方案集。
601D	8	上报方案	属性 2 (上报方案 report plan) ::= structure { 方案编号 unsigned, 上报通道 array OAD, 上报响应超时时间 TI, 最大重试次数 unsigned, 上报内容 structure { 类型 unsigned, 数据 Data } }

OI	IC	对象名称	实例的对象属性及方法定义									
			<pre> } 上报内容: <table border="1" data-bbox="754 383 1442 510"> <thead> <tr> <th>上报类型</th> <th>上报内容</th> <th>表示</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>OAD</td> <td>对象属性数据</td> </tr> <tr> <td>1</td> <td>RecordData</td> <td>上报记录型对象属性</td> </tr> </tbody> </table> RecordData ::= structure { 对象属性描述符 OAD, 记录型对象列选择 RCSD, 记录型对象行选择 RSD } </pre>	上报类型	上报内容	表示	0	OAD	对象属性数据	1	RecordData	上报记录型对象属性
上报类型	上报内容	表示										
0	OAD	对象属性数据										
1	RecordData	上报记录型对象属性										
6032	11	采集状态集	属性 2 ::= array 一个采集状态									
6033	8	一个采集状态	<pre> 一个采集状态 ::= structure { 通信地址 TSA, 中继级别 unsigned, 中继地址 TSA, 端口 OAD, 最后一次采集成功时间 date_time_s, 采集失败次数 unsigned, 相位 enum{未知 (0), A相 (1), B相 (2), C相 (3)}, 相序异常 enum{正常 (0), LN 互易 (1), 逆相序 (2)} } </pre>									
6034	11	采集任务监控集	属性 2 ::= array 采集任务监控单元									
6035	8	采集任务监控单元	<pre> 采集任务监控单元 ::= structure { 任务 ID unsigned, 任务执行状态 enum { 未执行 (0), 执行中 (1), 已执行 (2) }, 任务执行开始时间 date_time_s, 任务执行结束时间 date_time_s, 采集表计总数量 long-unsigned, 采集成功表计数量 long-unsigned, 已发送报文条数 long-unsigned, 已接收报文条数 long-unsigned } </pre> <p>说明：采集成功表计数量、已发送报文条数、已接收报文条数，每</p>									

OI	IC	对象名称	实例的对象属性及方法定义
			次任务执行先清零。
6040	8	采集启动时标	属性 2::=date_time_s
6041	8	采集成功时标	属性 2::=date_time_s
6042	8	采集存储时标	属性 2::=date_time_s
6050	8	线损特征字	属性 2（线损特征字）::=long-unsigned 特征字： D0-置 1 表示为台区总表 D1-置 1 表示为普通户表 D2-置 1 表示为分布式电源电表正向上网 D3-置 1 表示为分布式电源电表反向上网 D4-置 1 表示不参与线损计算，D5~D15-备用； 注：通过采集档案附属信息关联。
6051	6	线损计算数据	属性 2 ::= array 总及分相数值 总及分相数值 ::= structure <pre> { 供电量 double-long-unsigned, 售电量 double-long-unsigned, 倒送电量 double-long-unsigned, 分布式电源上网电量 double-long-unsigned, 线损率 long（换算：-2，单位：%）， 异常标志 enum { 正常 (0), 总表电量异常 (1), 户表电量异常 (2) } } </pre> 属性 4（线损统计周期）::= TI

A.9 OIA1=7H

OIA1=7H的对象标识定义见表A.9。

表 A.9 OIA1=7H 对象标识定义

OI	IC	对象名称	实例的对象属性及方法定义
7000	11	文件集合	方法 127: WriteFile (参数) 参数::=structure { 文件名 visible-string, 偏移 double-long-unsigned, 内容 octet-string } 写文件。 方法 128: Execute (文件名) 执行文件。 方法 129: DeleteFile (文件名) 文件名::=visible-string 删除文件。 方法 130: ReadFile (参数) 参数::=structure { 文件名 visible-string, 偏移 double-long-unsigned } 应答::=octet-string
7001	8	文件	属性 2: 文件 文件::=structure { 文件名 visible-string, 文件长度 long-unsigned, 创建时间 date_time_s, 修改时间 date_time_s } 文件长度: 单位字节。
7010	11	脚本集合	属性 2::=array 脚本 方法 127: Add (脚本) 添加更新一个脚本。 方法 128: Delete (脚本 id) 删除一个脚本。 方法 129: Execute (脚本 id)

OI	IC	对象名称	实例的对象属性及方法定义
			执行脚本。 方法 130: Clear (参数) 参数::=NULL 清空脚本集。 脚本 id::=long-unsigned
7011	8	脚本	属性 2::=脚本 脚本::=structure { 脚本 ID long-unsigned, 操作集 array 一个操作 } 一个操作::=APDU 一个操作等价于一个 APDU
7012	11	脚本执行结果集	属性 2::=array 一个脚本执行结果
7013	8	一个脚本执行结果	脚本执行结果::=structure { 脚本 ID long-unsigned, 脚本执行时间 date_time_s, 脚本执行结果集 array 一个执行结果 } 一个执行结果::=APDU 一个结果等价于一个 APDU
7100	11	扩展变量对象集合	属性 2::=扩展变量对象集合 扩展变量对象集合::=array 变量类对象 变量类对象::=Data
7101	11	扩展参变量对象集合	属性 2::=扩展参变量对象集合 扩展参变量对象集合::=array 参变量类对象 参变量类对象::=Data
7200	11	充电计划单集	属性 2 (充电计划单集合, 只读) ::=array 充电计划单单元 方法 127: 下发充电计划 (充电计划单) 返回结果::=structure { 充电申请单号 octet-string, 路由器资产编号 visible-string (SIZE(22)) }
7201	11	充电申请单集	属性 2 (充电申请单集合, 只读) ::=array 充电申请单单元 方法 127: 充电申请 (充电申请单) 返回结果::=structure { 充电申请单号 octet-string, 路由器资产编号 visible-string (SIZE(22)) }

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> } 方法 128: 结束充电订单 (参数) 参数 ::= structure { 充电申请单号 octet-string, 路由器资产编号 visible-string (SIZE(22)), 触发原因 enum { 用户结束 (1), 台区容量严重不足 (2), 路由器故障 (3), 充电桩故障 (4), 系统终止 (5) } } </pre>
7202	11	邀约对象集合	<p>属性 2 (邀约对象列表, 只读) ::= array 邀约对象单元</p> <p>方法 127: 配置 (array 邀约对象单元)</p> <p>批量添加或更新邀约对象单元。</p> <p>方法 128: 删除 (array 家电通信地址)</p> <p>家电设备地址 ::= octet-string</p> <p>方法 129: 清空 (参数)</p> <p>参数 ::= NULL</p>
7300	11	容器集合	<p>属性 2 (容器集, 只读) ::= array 容器单元</p> <p>方法 127: 创建 (参数)</p> <p>参数 ::= structure</p> <pre> { 容器名称 visible-string, 应用名称 visible-string, CPU 核数 unsigned, 内存限值 visible-string, 存储限值 visible-string, 挂载的宿主机文件夹 visible-string, 映射的硬件接口 visible-string, 应用下载文件及路径 visible-string } </pre> <p>方法 128: 修改配置 (参数)</p> <pre> { 容器名称 visible-string, CPU 核数 unsigned, 内存限值 visible-string, 存储限值 visible-string, </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			挂载的宿主机文件夹 visible-string, 映射的硬件接口 visible-string } 方法 129: 启动 (容器名称) 容器名称::=visible-string 方法 130: 停止 (容器名称) 容器名称::=visible-string 方法 131: 删除 (容器名称) 容器名称::=visible-string
7301	11	应用集合	属性 2 (应用集, 只读) ::=array 应用单元 方法 127: 安装 (参数) 参数::=structure { 容器名称 visible-string, 应用名称 visible-string, 安装包名称及路径 visible-string } 方法 128: 启动 (参数) 参数::=structure { 容器名称 visible-string, 应用名称 visible-string } 方法 129: 停止 (参数) 参数::=structure { 容器名称 visible-string, 应用名称 visible-string } 方法 130: 卸载 (参数) 参数::=structure { 容器名称 visible-string, 应用名称 visible-string } 方法 131: 更改应用资源配置 (参数) 参数::=structure { 容器名称 visible-string, 应用名称 visible-string, CPU 使用率限值 long-unsigned (单位: %, 换算: -2),

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> 内存使用率限值 long-unsigned (单位: %, 换算: -2) } 方法 132: 更改应用资源配置 (参数) 参数 ::= structure { 容器名称 visible-string, 应用名称 visible-string, 服务名称 visible-string, CPU 使用率限值 long-unsigned (单位: %, 换算: -2), 内存使用率限值 long-unsigned (单位: %, 换算: -2) } </pre>

A.10 01A1=8H

01A1=8H的对象标识定义见表A.10。

表 A.10 01A1=8H 对象标识定义

OI	IC	对象名称	实例的对象属性及方法定义
8000	8	远程控制	<p>属性 2 (配置参数)::=structure</p> <pre>{ 继电器拉闸电流门限值 double-long-unsigned(单位: A, 换算-4), 超电流门限延时时间 long-unsigned(单位: 分钟, 换算0) }</pre> <p>属性 4(告警状态, 只读)::=bit-string(SIZE(8)) 告警状态: bit0...bit7 分别按顺序对位表示 1...8 号继电器遥控告警输出状态, 置“1”: 处于告警状态; 置“0”: 处于非告警状态。</p> <p>属性 5 (命令状态, 只读)::=bit-string(SIZE(8)) 继电器命令状态: bit0...bit7 分别按顺序对位表示 1...8 号继电器遥控跳闸命令状态, 置“1”: 跳闸命令; 置“0”: 合闸命令。</p> <p>属性 6 (跳闸源模式字)::=bit-string(SIZE(16)) (无单位, 无换算) 跳闸源模式字中 bit0: 1 表示允许端子座过热跳闸, 0 表示禁止; bit1: 1 表示允许过载跳闸, 0 表示禁止; 其它位保留为 0。远程控制跳闸、本地费控跳闸不受跳闸源选项控制。</p> <p>属性 7 (友好用电参数)::= structure</p> <pre>{ 允许友好用电时长 long-unsigned(单位: 天, 无换算), 友好用电功率阈值 double-long(单位: kWh, 换算: -4), 过负荷判断延时时间 unsigned(单位: s, 无换算) }</pre> <p>友好用电功率阈值设置为 0 则不启用友好用电功能, 允许友好用电时长设置为 0 则允许功率小于阈值的情况下一直用电。友好用电功率阈值如果设置为 0 则不判断允许友好用电时长。</p> <p>方法 127: 触发告警 (参数) 参数::=NULL</p> <p>方法 128: 解除报警 (参数) 参数::=NULL</p> <p>方法 129: 跳闸 (参数) 参数::=array structure</p> <pre>{ 继电器 OAD, 告警延时 unsigned(单位: 分钟, 换算: 0), 限电时间 long-unsigned(单位: 分钟, 换算: 0; 值为 0 表示永久限电),</pre>

OI	IC	对象名称	实例的对象属性及方法定义
			<p>自动合闸 bool (True: 自动合闸; False: 非自动合闸)</p> <p>}</p> <p>方法 130: 合闸 (参数)</p> <p>参数 ::= array structure</p> <p>{</p> <p>继电器 OAD,</p> <p>命令 enum{合闸允许 (0), 直接合闸 (1)}</p> <p>}</p> <p>方法 131: 电能表明文合闸 (参数)</p> <p>参数 ::= array structure</p> <p>{</p> <p>继电器 OAD,</p> <p>命令 enum{合闸允许 (0), 直接合闸 (1)},</p> <p>密码 visible-string</p> <p>}</p>
8001	8	保电	<p>属性 2 (保电状态, 只读) ::= enum{解除 (0), 保电 (1), 自动保电 (2)}</p> <p>属性 3 ::= long-unsigned</p> <p>允许与主站最大无通信时长 (分钟), 0 表示不自动保电。</p> <p>属性 4 ::= long-unsigned</p> <p>上电自动保电时长 (分钟), 0 表示上电不自动保电。</p> <p>属性 5 ::= array 自动保电时段</p> <p>自动保电时段 ::= structure</p> <p>{</p> <p>起始时间 (时) unsigned,</p> <p>结束时间 (时) unsigned</p> <p>}</p> <p>时间段区间规则为前闭后开。</p> <p>方法 127: 投入保电 (参数)</p> <p>参数 ::= NULL</p> <p>禁止一切执行跳闸的继电器输出控制, 且恢复已跳闸的继电器输出控制。</p> <p>方法 128: 解除保电 (参数)</p> <p>参数 ::= NULL</p> <p>用于解除保电状态。</p> <p>方法 129: 解除自动保电 (参数)</p> <p>参数 ::= NULL</p> <p>用于解除自动保电状态。</p>
8002	8	催费告警	<p>属性 2 (催费告警状态, 只读) ::= enum{未告警 (0), 告警 (1)}</p> <p>属性 3 (催费告警参数, 只读) ::= structure</p> <p>{</p> <p>告警时段 octet-string (SIZE (3)),</p> <p>告警信息 visible-string (SIZE (1...200))</p> <p>}</p>

OI	IC	对象名称	实例的对象属性及方法定义
			} 告警时段: bit0~bit23 按顺序表示 0~23 点, 置 1 表示告警, 置 0 表示不告警。 方法 127: 催费告警投入 (催费告警参数) 催费告警参数同属性 3。 方法 128: 取消催费告警 (参数) 参数 ::= NULL
8003	11	一般中文信息	属性 2 ::= array ChineseInfo ChineseInfo ::= structure { 序号 unsigned, 发布时间 date_time_s, 已阅读标识 bool (True: 已阅读, False: 未阅读), 信息内容 visible-string(SIZE(1...200)) } 方法 127: 添加信息 (参数) 参数 ::= structure { 序号 unsigned, 发布时间 date_time_s, 信息内容 visible-string(SIZE(1...200)) } 方法 128: 删除信息 (序号) 序号参见方法 127。
8004	11	重要中文信息	属性 2 ::= array ChineseInfo ChineseInfo 定义参见 8003。 方法 127: 添加信息 (参数) 参数 ::= structure { 序号 unsigned, 发布时间 date_time_s, 信息内容 visible-string(SIZE(1...200)) } 方法 128: 删除信息 (序号) 序号参见方法 127。
8100	8	终端保安定值	终端保安定值 ::= long64 (单位: W, 换算: -1)
8101	8	终端功控时段	属性 2 (配置参数) ::= array unsigned 终端功控时段单元格式见表 A.11。
8102	8	功控告警时间	属性 2 (配置参数) ::= array unsigned 告警时间按顺序表示 1-n 轮次的功控告警时间 (单位: 分钟)
8103	13	时段功控	属性 2 (控制方案集) ::= array 时段功控配置单元

OI	IC	对象名称	实例的对象属性及方法定义
			方法 127：时段功控方案切换（参数） 参数 ::= structure { 总加组对象 OI, 控制方案 structure { 时段功控投入标识 bit-string(SIZE(8)), 时段功控定值方案号 unsigned } } 时段功控投入标识：D0...D7 按顺序对位表示第 1...第 8 时段，置“1”：有效，置“0”：无效。 时段功控定值方案号：数值范围 0...2 依次表示第 1...第 3 套方案。
8104	13	厂休控	属性 2（控制方案集） ::= array 厂休控配置单元
8105	13	营业报停控	属性 2（控制方案集） ::= array 营业报停控配置单元
8106	13	当前功率下浮控	属性 2：不可访问 方法 127 投入（参数） 参数 ::= structure { 总加组对象 OI, 控制方案 structure { 当前功率下浮控定值滑差时间 unsigned（单位：分钟）， 当前功率下浮控定值浮动系数 integer（单位：%）， 控后总加有功功率冻结延时时间 unsigned（单位：分钟）， 当前功率下浮控的控制时间 unsigned（单位：0.5 小时）， 当前功率下浮控第 1 轮告警时间 unsigned（单位：分钟）， 当前功率下浮控第 2 轮告警时间 unsigned（单位：分钟）， 当前功率下浮控第 3 轮告警时间 unsigned（单位：分钟）， 当前功率下浮控第 4 轮告警时间 unsigned（单位：分钟） } } 当前功率下浮控定值浮动系数：负值表示下浮，正值表示上浮。
8107	13	购电控	属性 2（控制方案集） ::= array 购电控配置单元
8108	13	月电控	属性 2（控制方案集） ::= array 月电控配置单元
8109	8	时段功控配置单元	属性 2 ::= structure { 总加组对象 OI, 方案标识 bit-string(SIZE(8)), 第一套定值 PowerCtrlParam, 第二套定值 PowerCtrlParam, }

OI	IC	对象名称	实例的对象属性及方法定义
			第三套定值 PowerCtrlParam, 时段功控定值浮动系数 integer (单位: %) } 方案标识: bit0...bit2 按顺序对位表示第 1...第 3 套定值, 置“1”: 有效, 置“0”: 无效。 PowerCtrlParam ::= structure { 时段号 bit-string(SIZE(8)), 时段 1 功控定值 long64 (单位: W, 换算: -1), 时段 2 功控定值 long64 (单位: W, 换算: -1), 时段 3 功控定值 long64 (单位: W, 换算: -1), 时段 4 功控定值 long64 (单位: W, 换算: -1), 时段 5 功控定值 long64 (单位: W, 换算: -1), 时段 6 功控定值 long64 (单位: W, 换算: -1), 时段 7 功控定值 long64 (单位: W, 换算: -1), 时段 8 功控定值 long64 (单位: W, 换算: -1) } 时段号: bit0...bit7 按顺序对位表示第 1...第 8 时段, 置“1”: 有效, 置“0”: 无效。
810A	8	厂休控配置单元	属性 2 ::= structure { 总加组对象 OI, 厂休控定值 long64 (单位: W, 换算: -1), 限电起始时间 date_time_s (年=FFFFH, 月=FFH, 日=FFH), 限电延续时间 long-unsigned (单位: 分钟), 每周限电日 bit-string(SIZE(8)) } 每周限电日: D1...D7 表示星期一...星期日, D0=0。
810B	8	营业报停控配置单元	属性 2 ::= structure { 总加组对象 OI, 报停起始时间 date_time_s (时=FFH, 分=FFH), 报停结束时间 date_time_s (时=FFH, 分=FFH), 报停控功率定值 long64 (单位: W, 换算: -1) }
810C	8	购电控配置单元	属性 2 ::= structure { 总加组对象 OI, 购电单号 double-long-unsigned, 追加/刷新标识 enum{追加 (0), 刷新 (1)}, 购电类型 enum{电量 (0), 电费 (1)}, }

OI	IC	对象名称	实例的对象属性及方法定义
			购电量(费)值 long64 (单位: kWh/元, 换算: -4), 报警门限值 long64 (单位: kWh/元, 换算: -4), 跳闸门限值 long64 (单位: kWh/元, 换算: -4), 购电控模式 enum{本地模式 (0), 远程模式 (1)} }
810D	8	月电控配置单元	属性 2 ::= structure { 总加组对象 OI, 月电量控定值 long64 (单位: kWh, 换算: -4), 报警门限值系数 unsigned (单位: %), 月电量控定值浮动系数 integer (单位: %) }
810E	8	控制对象	
810F	8	跳闸轮次	
8110	8	电控定值	
8200	8	合分遥控	数值 ::= enum{分 (1), 合 (2)}; 单位: 无, 换算: 无
8201	8	遥控软压板投退	数值 ::= enum{分 (1), 合 (2)}; 单位: 无, 换算: 无
8202	8	电池活化/退出	数值 ::= enum{分 (1), 合 (2)}; 单位: 无, 换算: 无
8203	8	信号复归	数值 ::= enum{分 (1), 合 (2)}; 单位: 无, 换算: 无
8204	8	解除闭锁	数值 ::= enum{分 (1), 合 (2)}; 单位: 无, 换算: 无

终端功控时段数据单元格式见表A.11。其中：

- 每半小时以两位编码表示 4 种控制状态：取值 0...3 依次表示不控制、控制 1、控制 2、保留。
- 控制状态标识的应用规则：连续时间单元具有相同控制状态标识表示同一控制时段；连续时间单元具有不同控制状态标识，表示相邻的两个时段，控制 1 与控制 2 用以区分具备 2 个不同的定值的连续时段，当控制状态标识发生变化时，表示前一控制时段结束，后一控制时段开始，对于不连续的控制时段可以用控制 1 或控制 2 表示。

表 A.11 终端功控时段数据单元格式

数据内容								字节数
D7	D6	D5	D4	D3	D2	D1	D0	
1: 30-2: 00		1: 00-1: 30		0: 30-1: 00		0: 00-0: 30		第 1 字节
3: 30-4: 00		3: 00-3: 30		2: 30-3: 00		2: 00-2: 30		第 2 字节
.....	
23: 30-24: 00		23: 00-23: 30		22: 30-23: 00		22: 00-22: 30		第 12 字节

A.11 OIA1=FH

OIA1=FH的对象标识定义见表A.12。

表 A.12 OIA1=FH 对象标识定义

OI	IC	对象名称	实例的对象属性及方法定义
F000	18	文件分帧传输管理	属性4（文件内容） ::=octet-string 传输文件的内容，目标文件不存在则创建，并将当前偏移位置清零。 属性5（当前偏移位置） ::=double-long-unsigned 当前传输文件的所在偏移位置，单位byte。 方法127：文件下载（参数） 参数 ::=structure { 文件路径 visible-string, 偏移位置 double-long-unsigned, 读取长度 double-long-unsigned } 应答 ::=octet-string
F001	18	文件分块传输管理	属性4（传输块状态字，只读） ::=bit-string 按 bit 位标识每个数据块的传输状态。bitN=0，表示未传输，bitN=1，表示传输成功（N从0...总传输块数-1）。 属性5（下载方的标识，只读） ::=octet-string 属性6（支持传输块大小，只读） ::=long-unsigned 属性7（执行升级时间，只读） ::=date_time_s 方法7：启动传输（参数） 参数 ::=structure { 文件信息 structure, 传输块大小 long-unsigned, 校验 structure } 方法8：写文件（参数） 参数 ::=structure { 块序号 long-unsigned, 块数据 octet-string } 方法9：读文件（参数） 参数 ::=structure { 块序号 long-unsigned }

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> 应答 ::= structure { 块数据 octet-string } 方法 10: 软件比对 (参数) 参数 ::= structure { CPU 编号 unsigned, 密钥索引 unsigned, 因子起始地址 double-long-unsigned, 数据起始地址 double-long-unsigned, 待加密数据长度 long-unsigned } 应答 ::= structure { 比对块数据 octet-string } </pre> <p>软件比对是指：对设备的软件进行比对，命令中 CPU 编号的 bit0..bit2 有效，其它保留。该字节缺省为 00，如设备内部存在多个 CPU，主 CPU 编号为 0，其它自行编号，最多支持 8 个 CPU。</p> <p>软件比对命令中如果比对因子起始地址或比对数据起始地址超出设备 MCU 的地址空间，则认为设备不支持这部分数据，返回应答“地址异常”。</p> <p>软件比对命令中比对因子和比对数据的起始地址用绝对地址表示。软件比对命令中嵌有安全模块的设备应采用安全模块加密保护方式比对，不支持异或加密方式比对；未嵌安全模块的设备应采用异或加密方式比对。</p> <p>软件比对命令中未嵌安全模块的设备比对密钥索引固定为 0。异或加密方式见附录 E。</p> <p>方法 13: 执行升级 (参数) ::= date_time_s 当日期为全 FF 是表示立即执行升级。</p> <p>方法 14: 启动传输 (参数) 参数 ::= structure { 文件信息 structure, 传输块大小 long-unsigned, 校验 structure, 兼容软件版本号 array visible-string, 兼容硬件版本号 array visible-string, 下载方的标识 octet-string }</p>

OI	IC	对象名称	实例的对象属性及方法定义
			文件信息数据结构见表 185 属性 2； 传输块大小单位：byte； 校验 ::= structure { 校验类型 enum { CRC 校验（默认） (0)， md5 校验 (1)， SHA1 校验 (2)， 其他 (255) } 校验值 octet-string }
F002	18	文件扩展传输管理	属性 4（服务器信息） ::= structure { IP 地址 octet-string， 端口 long-unsigned， 用户名 visible-string， 密码 visible-string } 扩展传输是对基于 TCP 连接的通用文件传输协议的扩展支持。 方法 7：从服务器下载（参数） 参数 ::= structure { 文件信息 structure， 协议类型 enum { telnet+zmodem 协议 (0)， ftp 协议 (1)， sftp 协议 (2)， http 协议 (3)， https 协议 (4) } } 以客户机模式主动连接指定远程服务器下载文件，并通过“文件传输状态”反馈执行情况。目标文件不存在则创建。 方法 8：上传到服务器（参数） 参数 ::= structure { 文件信息 structure， 协议类型 enum

OI	IC	对象名称	实例的对象属性及方法定义
			<p>} 以客户机模式主动连接指定远程服务器上传文件，并通过“命令结果”反馈执行情况。源文件不存在则返回错误，目标文件不存在则创建。 文件信息数据结构见表 185 属性 2。</p>
F100	21	ESAM	<p>属性 16（白名单） ::=array 设备序列号 设备序列号 ::= octet-string 属性 17（终端地址） ::=octet-string</p>
F101	8	安全模式参数	<p>属性 2（安全模式选择） ::=enum { 不启用安全模式（0）， 启用安全模式（1） } 属性 3（显式安全模式参数） ::=array 安全模式参数 安全模式参数 ::=structure { 对象标识 OI， 安全模式 long-unsigned } 显式安全模式参数（设置值），如果对象安全性不在属性 3 中，按默认安全模式参数执行。如果同一对象安全性在显示安全模式参数、默认安全模式参数中均有说明，则按显式安全模式参数执行。默认安全参数定义见附录 F。 属性 4（SAL 安全应用数据链路层参数） ::=enum { 不启用 SAL（0）， 启用 SAL（1） } 方法 1：复位（参数） 参数 ::=integer（0） 复位时，清空属性 3。 方法 127：增加显式安全对象（参数） 参数 ::=structure { 对象标识 OI， 权限 long-unsigned } 方法 128：删除显式安全模式（对象标识） 对象标识 ::=OI 方法 129：批量增加显式安全对象（array 安全对象） 安全对象 ::=structure</p>

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> { 对象标识 OI, 安全模式 long-unsigned } </pre>
F200	22	RS232	<pre> 属性 2（设备对象列表，只读） ::=array 端口 端口 ::=structure { 端口描述符 visible-string, 端口参数 COMDCB, 端口功能 enum{远程通信（0），抄表（1），停用（3）} } 方法 127：配置端口（参数） 参数 ::=structure { 端口号 OAD, 端口参数 COMDCB, 端口功能 enum } </pre>
F201	22	RS485	同 F200。
F202	22	红外	<pre> 属性 2（设备对象列表，只读） ::=array 红外端口 红外端口 ::=structure { 端口描述符 visible-string, 端口参数 COMDCB } 方法 127：配置端口（参数） 参数 ::=structure { 端口号 OAD, 端口参数 COMDCB } </pre>
F203	22	开关量输入	<pre> 属性 2（设备对象列表，只读） ::=array 开关量单元 开关量单元 ::=structure { 状态 ST unsigned, 变位 CD unsigned } 状态 ST——0：“分”状态；1：“合”状态。 变位 CD——0：自前次遥信传送后无状态变化；1：自前次遥信传 送后至少有一次状态变化。 属性 4 ::=structure { </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			开关量接入标志 bit-string(SIZE (8)), 开关量属性标志 bit-string(SIZE (8)) } 开关量接入标志: bit0...bit7 按顺序对位表示第 1...8 路状态量输入, 置“1”: 接入, 置“0”: 未接入。 开关量属性标志: bit0...bit7 按顺序对位表示第 1...8 路状态量输入, 置“1”: 常开 触点, 置“0”: 常闭触点。
F204	22	直流模拟量	属性 2 (设备对象列表, 只读) ::=array 直流模拟量 直流模拟量::=double-long 属性 4::=array 直流模拟量配置 直流模拟量配置::=structure { 量程起始值 double-long, 量程结束值 double-long, 换算及单位 Scaler_Unit }
F205	22	继电器输出	属性 2 (设备对象列表, 只读) ::=array 继电器单元 继电器单元::=structure { 描述符 visible-string, 当前状态 enum{合闸 (0), 跳闸 (1)}, 开关属性 enum{脉冲式 (0), 保持式 (1)}, 接线状态 enum{接入 (0), 未接入 (1)} } 方法 127: 修改开关属性 (参数) 参数::=structure { 继电器号 OAD, 开关属性 enum }
F206	22	告警输出	属性 2 (设备对象列表, 只读) ::=array 告警输出 告警输出::=enum { 未输出 (0), 输出 (1) } 属性 4::=array 允许告警时段 允许告警时段::=structure { 起始时间 Time, }

OI	IC	对象名称	实例的对象属性及方法定义
			结束时间 Time }
F207	22	多功能端子	属性 2（设备对象列表，只读） ::=array 端子功能 端子功能 ::=enum { 秒脉冲输出 (0), 需量周期 (1), 时段投切 (2), 正向谐波脉冲 (3), 反向谐波脉冲 (4), 无功脉冲 (5), 关闭 (255) } 方法 127: 修改工作模式（参数） 参数 ::=structure { 路号 OAD, 端子功能 enum }
F208	22	交采接口	属性 2（设备对象列表，只读） ::=array 交采单元 交采单元 ::=structure { 交采描述符 visible-string }
F209	22	载波/微功率无线接口	属性 2（设备对象列表，只读） ::=array 本地通信模块单元 本地通信模块单元 ::=structure { 端口描述符 visible-string, 通信参数 COMDCB, 版本信息 VersionInfo } VersionInfo ::=structure { 厂商代码 visible-string(SIZE(2)), 芯片代码 visible-string(SIZE(2)), 版本日期 Date, 软件版本 long-unsigned } 属性 5（从节点对象列表，只读） ::=array 从节点单元 属性 6（从节点对象列表更新周期） ::=TI 属性 7（网络拓扑信息，只读） ::=array 网络拓扑单元

OI	IC	对象名称	实例的对象属性及方法定义
			<p>网络拓扑单元 ::= structure</p> <pre>{ 节点地址 octet-string(SIZE(6)), 节点类型 enum{主节点(0), 从节点(1)}, 节点标识 long-unsigned, 代理节点标识 long-unsigned, 节点信息 unsigned }</pre> <p>节点标识: 本站点的节点标识 (TEI); 代理节点标识: 本站点的代理站点节点标识 (TEI); 节点信息: bit0~bit3 位, 节点层级, 本站点的网络层级, 0 级代表 0 层级, 依次类推; bit4~bit7 位; 节点角色, 本站点的网络角色, 0x0: 无效, 0x1: 末梢节点 (STA), 0x2: 代理节点 (PCO), 0x3: 保留, 0x4: 主节点 (CC0)。</p> <p>属性 8 (多网信息, 只读) ::= structure</p> <pre>{ 本节点网络标识号 octet-string(SIZE(3)), 本节点主节点地址 octet-string(SIZE(6)), 邻居节点网络标识号 array octet-string(SIZE(3)) }</pre> <p>属性 9 (宽带载波频段序号) ::= unsigned 属性 10 (相位信息, 只读) ::= array 节点相位信息单元 属性 11 (台区识别启停标记) ::= unsigned</p> <p>启动标志: bit0: 台区识别使能标志 (0: 禁止; 1: 允许) bit1: 台区改切使能标志 (0: 禁止; 1: 允许) bit2: 离网感知使能标志 (0: 禁止; 1: 允许) bit3~bit7: 备用。</p> <p>属性 12 (拒绝从节点入网列表) ::= array octet-string 属性 13 (拒绝节点上报使能标志) ::= enum{禁止 (0), 使能 (1)}</p> <p>属性 20 (网络规模, 只读) ::= unsigned</p> <p>方法 127: 透明转发 (参数)</p> <p>参数 ::= structure</p> <pre>{ 通信地址 TSA, 接收等到报文超时时间 (秒) long-unsigned, 透明转发命令 octet-string }</pre> <p>返回结果 ::= octet-string</p> <p>方法 128: 配置端口参数 (参数)</p> <p>参数 ::= structure</p>

OI	IC	对象名称	实例的对象属性及方法定义
			<pre> { 端口号 OAD, 通信参数 COMDCB } 方法 140: 允许入网 (参数) 参数 ::= array 允许入网从节点信息 允许入网从节点信息 ::= structure { 从节点通信地址 octet-string, 芯片管理 ID octet-string } 方法 141: 配置从节点主动上报 (参数) 参数 ::= structure { 上报类型 enum{全网上报 (0), 单点上报 (1)}, 从节点地址列表 array octet-string, 允许数据上报 bool, 持续时间 double-long-unsigned (单位: min, 换算: 0) } </pre>
F20A	22	脉冲输入设备	<p>属性 2 (设备对象列表, 只读) ::= array 脉冲输入端口描述符 脉冲输入端口描述符 ::= visible-string</p>
F20B	22	蓝牙	<p>属性 2 (设备对象列表, 只读) ::= array 蓝牙模块 蓝牙模块 ::= structure</p> <pre> { 模块序号 unsigned, 蓝牙信息 BluetoothInfo, 配对密码 visible-string, 配对模式 enum { Numeric Comparison (0), Just Works (1), Passkey Entry (2), Out of Band (3) } } </pre> <p>属性 4 (从设备列表) ::= array 蓝牙从设备 属性 5 (连接信息, 只读) ::= array structure</p> <pre> { 端口号 OAD, 连接的主设备 array BluetoothInfo, 连接的从设备 array BluetoothInfo, </pre>

OI	IC	对象名称	实例的对象属性及方法定义
			<p>连接状态 enum(连接断开 (0), 连接成功 (1))</p> <p>}</p> <p>属性 6 (工作参数) ::= array structure</p> <p>{</p> <p> 端口号 OAD,</p> <p> 发射功率档 unsigned,</p> <p> 广播间隔 long-unsigned (单位: ms, 无换算),</p> <p> 扫描间隔 long-unsigned (单位: ms, 无换算)</p> <p>}</p> <p>广播间隔范围: 40ms~1000ms, 默认 40ms;</p> <p>扫描间隔范围: 55ms~110ms, 默认 100ms。</p> <p>方法 1: 复位 (参数)</p> <p>参数 ::= OAD (端口号)</p> <p>方法 127: 配置端口 (参数)</p> <p>参数 ::= structure</p> <p>{</p> <p> 端口号 OAD,</p> <p> 蓝牙信息 BluetoothInfo</p> <p>}</p> <p>BluetoothInfo ::= structure</p> <p>{</p> <p> 蓝牙描述符 visible-string,</p> <p> MAC 地址 octet-string</p> <p>}</p> <p>蓝牙从设备 ::= structure</p> <p>{</p> <p> MAC 地址 octet-string,</p> <p> 资产号 octet-string</p> <p>}</p> <p>方法 128: 启动误差检定 (参数)</p> <p>参数 ::= structure</p> <p>{</p> <p> 端口号 OAD,</p> <p> 脉冲类型 enum</p> <p> {</p> <p> 有功 (0),</p> <p> 无功 (1),</p> <p> 正向谐波 (2),</p> <p> 反向谐波 (3),</p> <p> 时钟 (4)</p> <p> },</p> <p>}</p>

OI	IC	对象名称	实例的对象属性及方法定义
			射频调制模式 enum{ BLE 1Mbps (1), BLE 2Mbps (2) }, 发射功率 enum{0 档(0), 1 档(1), 2 档(2), 3 档(3), 4 档(4), 5 档(5) }, 信道频率 long-unsigned (单位: MHz, 无换算), 蓝牙主机 MAC 地址 octet-string, 延时 long-unsigned (单位: us, 无换算) } 方法 129: 退出误差检定 (参数) 参数 (端口号) ::=OAD 方法 130: 启动连接 (参数) 参数::=structure { 端口号 OAD, 蓝牙主机 MAC 地址 octet-string, 蓝牙从机 MAC 地址 octet-string }
F20C	22	230M 无线专网接口	属性 2 (设备对象列表, 只读) ::=array 230 无线专网模块 230 无线专网模块::=structure { 端口描述符 visible-string, } 属性 4::=array 频道设置 频道设置 ::=unsigned (取值范围 1...16) 属性 5 (只读) ::=array 有效信号强度 有效信号强度::= integer (单位 dBμV)
F20D	22	模组	属性 2 (对象列表, 只读) ::=array 模组信息 模组信息::=structure { 槽位号 unsigned, 模块设备型号 visible-string, 模块设备 ID visible-string, 软件版本 double-long, 软件发布日期 visible-string, 硬件版本 double-long, 硬件发布日期 visible-string, 厂商编码 visible-string, 通道配置 array 通道信息 } 通道信息::= structure {

OI	IC	对象名称	实例的对象属性及方法定义
			接口类型 unsigned, 功能配置 unsigned }
F20E	22	USB	属性 2 (对象列表, 只读) ::=array USB 设备 USB 设备 ::=structure { 端口描述符 visible-string }
F20F	22	服务组件	属性 2 (对象列表, 只读) ::=array 组件 组件 ::=structure { 组件名称 visible-string, 版本信息 double-long-unsigned, 发布日期 date_time_s }
F210	8	从节点单元	属性 2 ::=structure { 从节点序号 long-unsigned, 从节点通信地址 octet-string, 从节点描述符 visible-string, }
F211	8	网络拓扑单元	属性 2 ::=structure { 节点地址 octet-string(SIZE(6)), 节点类型 enum{主节点(0), 从节点(1)}, 节点标识 long-unsigned, 代理节点标识 long-unsigned, 节点信息 unsigned } 节点标识: 本站点的节点标识 (TEI); 代理节点标识: 本站点的代理站点节点标识 (TEI); 节点信息: bit0~bit3 位, 节点层级, 本站点的网络层级, 0 级代表 0 层级, 依次类推; bit4~bit7 位, 节点角色, 本站点的网络角色, 0x0: 无效, 0x1: 末梢节点 (STA), 0x2: 代理节点 (PC0), 0x3: 保留, 0x4: 主节点 (CC0)。
F212	8	节点相位信息单元	属性 2 (节点相位信息单元) ::=structure { 节点序号 long-unsigned, 节点通信地址 octet-string(SIZE(6)), 节点相位信息 long-unsigned }

OI	IC	对象名称	实例的对象属性及方法定义																																
			<p>节点相位信息：</p> <p>bit0~bit2：分别表示 A、B、C 相</p> <p>bit3：表计类型 0（单相表）、1（三相表）</p> <p>bit4：线路异常 0（无异常）、1（有异常，单相表表示 LN 互易，三相表表示逆相序）</p> <p>bit5~bit7：三相表相序排列</p> <table border="1" data-bbox="754 562 1369 949"> <thead> <tr> <th>bit7</th> <th>bit6</th> <th>bit5</th> <th>相序</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>ABC</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>ACB</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>BAC</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>BCA</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>CAB</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>CBA</td> </tr> <tr> <td colspan="3">其他</td> <td>保留</td> </tr> </tbody> </table>	bit7	bit6	bit5	相序	0	0	0	ABC	0	0	1	ACB	0	1	0	BAC	0	1	1	BCA	1	0	0	CAB	1	0	1	CBA	其他			保留
bit7	bit6	bit5	相序																																
0	0	0	ABC																																
0	0	1	ACB																																
0	1	0	BAC																																
0	1	1	BCA																																
1	0	0	CAB																																
1	0	1	CBA																																
其他			保留																																
F213	22	对象容器集合	<p>属性 2（对象容器列表，只读） ::=array 对象容器</p> <p>对象容器 ::=array OAD</p> <p>属性 5（对象容器数据列表） ::=array 对象容器数据</p> <p>对象容器数据 ::=structure</p> <pre data-bbox="754 1167 1075 1339"> { 第 1 个对象属性数据 Data, ... 第 n 个对象属性数据 Data } </pre> <p>方法 127：添加（参数）</p> <p>参数 ::=structure</p> <pre data-bbox="754 1435 1075 1585"> { 对象容器 OAD, 对象属性列表 array OAD } </pre> <p>方法 128：删除（参数）</p> <p>参数 ::=structure</p> <pre data-bbox="754 1682 1075 1832"> { 对象容器 OAD, 对象属性列表 array OAD } </pre> <p>方法 129：清除（参数）</p> <p>参数 ::=structure</p> <pre data-bbox="754 1928 938 2002"> { 对象容器 OAD } </pre>																																

OI	IC	对象名称	实例的对象属性及方法定义
			}
F300	17	自动轮显	
F301	17	按键轮显	
F302	8	图像	<p>属性 2（图像信息） ::= array structure</p> <pre>{ 起始行坐标 long-unsigned, 起始列坐标 long-unsigned, 图像行坐标像素长度 long-unsigned, 图像列坐标像素长度 long-unsigned, 图像数据 array bit-string, 图像数据取模方式 enum { 逐列式 (0), 逐行式 (1), 行列式 (2), 列行式 (3) } }</pre> <p>属性 3（图像控制信息，只读） ::= structure</p> <pre>{ 图像数据显示方式 enum{ 阴码 (0), 阳码 (1) }, 图像数据取模方式 enum { 逐列式 (0), 逐行式 (1), 行列式 (2), 列行式 (3) }, 当前图像个数 unsigned }</pre> <p>方法 1：复位（参数） 参数 ::= integer (0) 清零当前所有图像。 注：图像数据采用 array bit-string 类型，具体如下： 每一列构成一条 bit-string，其 bit-string 长度为行数； array bit-string 的数组元素个数为列数。 液晶原点位于电表的左上方，向下行数增加，向右列数增加。</p>
F400	8	日志管理	<p>属性 2（日志文件列表，只读） ::= array 日志文件信息</p> <p>日志文件信息 ::= structure</p> <pre>{ 日志文件名称及路径 visible-string,</pre>

OI	IC	对象名称	实例的对象属性及方法定义
			日志文件大小 double-long-unsigned (单位: KB, 换算: 0), 日志文件创建时间 datetime_s } 方法 127: 打包上传日志 (日志文件名称及路径) 日志文件名称及路径 ::= visible-string 方法 128: 删除日志文件 (日志文件名称及路径) 日志文件名称及路径 ::= visible-string 方法 129: 清除全部日志 (参数) 参数 ::= NULL
F401	8	容器单元	属性 2 (容器单元) ::= structure { 容器 ID unsigned, 容器名称 visible-string, 容器状态 visible-string, CPU 核数 unsigned, 内存限值 visible-string, 存储限值 visible-string, 挂载的宿主机文件夹 visible-string, 映射的硬件接口 visible-string, 应用下载文件及路径 visible-string, 容器 CPU 使用率 long-unsigned (单位: %, 换算: -2), 容器内存使用率 long-unsigned (单位: %, 换算: -2), 容器存储使用率 long-unsigned (单位: %, 换算: -2), 容器 IP 地址 octet-string, 容器镜像名称 visible-string, 容器创建时间 datetime_s, 容器启动时间 datetime_s, 容器运行时间 double-long-unsigned (单位: 秒, 换算: 0) }
F402	8	应用单元	属性 2 (应用单元) ::= structure { 应用序号 unsigned, 应用名称 visible-string, 容器名称 visible-string, 应用版本 visible-string, 应用哈希值 visible-string, 应用服务列表 array 应用服务信息 } 应用服务信息 ::= structure {

OI	IC	对象名称	实例的对象属性及方法定义
			服务索引 unsigned, 服务名称 visible-string, 服务是否使能 bool, 服务状态 enum{运行 (0), 停止 (1)}, CPU 使用率限值 long-unsigned (单位: %, 换算: -2), CPU 实际使用率 long-unsigned (单位: %, 换算: -2), 内存实际使用率 long-unsigned (单位: %, 换算: -2), 启动时间 datetime_s }

B

附 录 B
(规范性附录)
物理单位枚举定义及数值举例

B.1 物理单位枚举定义见表A.13。

表 A.13 物理单位枚举定义

代码	单位	量	单位名称	SI 定义
1	a	时间	年	
2	mo	时间	月	
3	wk	时间	周	7*24*60*60s
4	d	时间	日	24*60*60s
5	h	时间	小时	60*60s
6	min.	时间	分	60s
7	s	时间 (t)	秒	s
8	°	角度	度	rad*180/π
9	°C	温度 (T)	摄氏度	K-273.15
10	CNY	(当地) 货币		
11	m	长度 (l)	米	m
12	m/s	速度 (v)	米/秒	m/s
13	m ³	体积 (V) r _v , 仪表常数或脉冲值 (体积)	立方米	m ³
14	m ³	修正的体积	立方米	m ³
15	m ³ /h	流量	立方米/小时	m ³ /(60*60s)
16	m ³ /h	修正的流量	立方米/小时	m ³ /(60*60s)
17	m ³ /d	流量	立方米/天	m ³ /(24*60*60s)
18	m ³ /d	修正的流量	立方米/天	m ³ /(24*60*60s)
19	l	容积	升	10 ⁻³ m ³
20	kg	质量 (m)	千克	kg
21	N	力 (F)	牛顿	N
22	Nm	扭矩	牛顿米	J=Nm=Ws
23	Pa	压强 (p)	帕斯卡	N/m ²
24	bar	压力 (p)	巴	10 ⁻⁵ N/m ²
25	J	能量	焦耳	J=Nm=Ws
26	J/h	热功	焦每小时	J/(60*60s)
27	W	有功功率 (P)	瓦	W=J/s
28	kW	有功功率 (P)	千瓦	kW=J/(s*1000)
29	VA	视在功率 (S)	伏安	
30	kVA	视在功率 (S)	千伏安	

代码	单位	量	单位名称	SI 定义
31	var	无功功率 (Q)	乏	
32	kvar	无功功率 (Q)	千乏	
33	kWh	有功能量 r_w , 有功电能表常数或脉冲值	千瓦·时	kW*(60*60s)
34	kVAh	视在能量 r_s , 视在电能表常数或脉冲值	千伏·安·小时	kVA*(60*60s)
35	kvarh	无功能量 r_b , 无功电能表常数或脉冲	千乏·时	kvar*(60*60s)
36	A	电流 (I)	安培	A
37	C	电量 (Q)	库仑	C=As
38	V	电压 (U)	伏特	V
39	V/m	电场强度 (E)	伏/米	V/m
40	F	电容 (C)	法拉	C/V=As/V
41	Ω	电阻 (R)	欧姆	$\Omega = V/A$
42	$\Omega m^2/m$	电阻率 (ρ)	欧姆米	Ωm
43	Wb	磁通量 (Φ)	韦伯	Wb=Vs
44	T	磁通密度 (T)	特斯拉	Wb/m ²
45	A/m	磁场强度 (H)	安培/米	A/m
46	H	电感 (L)	亨利	H=Wb/A
47	Hz	频率	赫兹	1/s
48	1/(Wh)	有功能量表常数或脉冲		imp/kWh
49	1/(varh)	无功能量表常数或脉冲		
50	1/(VAh)	视在能量表常数或脉冲		
51	%	百分比	百分之	
52	byte	字节	字节	
53	dBm	分贝毫瓦		
54	CNY/kWh	电价		
55	Ah	安时	安时	
56	ms	毫秒	毫秒	
57	db μ V	发射电平		
...	保留			
254	其他单位			
255	无单位、缺单位、计数			

B.2 数值举例见表A.14。

表 A.14 数值举例

数值	换算	单位	数据
263783	-4	kWh	26.3783 kWh
60201	-3	A	60.201 A

数值	换算	单位	数据
22000	-2	V	220 V
1	-1	°C	0.1 °C
3467	0	V	3467 V
593	3	Wh	593 kWh

B A

附 录 C
(规范性附录)
有关一致性协商

C.1 协议一致性协商

协议一致性协商内容定义见表B.1。

表 B.1 协议一致性协商内容定义

数据类型定义	说明
<pre> ProtocolConformance ::= bit-string (SIZE (64)) { 应用连接协商 Application Association (0), 请求对象属性 Get Normal (1), 批量请求基本对象属性 Get With List (2), 请求记录型对象属性 Get Record (3), 代理请求对象属性 Get Proxy (4), 代理请求记录型对象属性 Get Proxy Record (5), 请求分帧后续帧 Get Subsequent Frame (6), 设置基本对象属性 Set Normal (7), 批量设置基本对象属性 Set With List (8), 设置后读取 Set With Get (9), 代理设置对象属性 Set Proxy (10), 代理设置后读取对象属性 Set Proxy With Get (11), 执行对象方法 Action Normal (12), 批量执行对象方法 Action With List (13), 执行方法后读取 Action With List (14), 代理执行对象方法 Action Proxy (15), 代理执行后读取 Action Proxy With Get (16), 事件主动上报 Active Event Report (17), 事件尾随上报 (18), 事件请求访问位 ACD 上报 (19), 分帧数据传输 (20), 安全方式传输 (31), 对象属性 ID 为 0 的读取访问 (32), 对象属性 ID 为 0 的设置访问 (33) } </pre>	

C.2 功能一致性协商

功能一致性协商内容定义见表B.2。

表 B.2 功能一致性协商内容定义

数据类型定义	说明
<pre>FunctionConformance ::= bit-string (SIZE (128)) { 电能量计量 (0), 双向有功计量 (1), 无功电能计量 (2), 视在电能计量 (3), 有功需量 (4), 无功需量 (5), 视在需量 (6), 复费率 (7), 阀控 (8), 本地费控 (9), 远程费控 (10), 基波电能 (11), 谐波电能 (12), 谐波含量 (13), 波形失真度 (14), 多功能端子输出 (15), 事件记录 (16), 事件上报 (17), 温度测量 (18), 状态量监测 (19), 以太网通信 (20), 红外通信 (21), 蓝牙通信 (22), 多媒体采集 (23), 保留 (24), 直流模拟量 (25), 弱电端子 12V 输出 (26), 搜表 (27), 三相负载平衡 (28), 升级 (29) }</pre>	

C B

附 录 D
 (资料性附录)
 校验算法

D.1 校验算法

```

/*
 * ul6 represents an unsigned 16-bit number. Adjust the typedef for
 * your hardware.
 * Drew D. Perkins at Carnegie Mellon University.
 * Code liberally borrowed from Mohsen Banan and D. Hugh Redelmeier.
 */
typedef unsigned short ul6;
/*
 * FCS lookup table as calculated by the table generator.
 */
static ul6 fcstab[256]={
    0x0000, 0x1189, 0x2312, 0x329b, 0x4624, 0x57ad, 0x6536, 0x74bf,
    0x8c48, 0x9dc1, 0xaf5a, 0xbed3, 0xca6c, 0xdbe5, 0xe97e, 0xf8f7,
    0x1081, 0x0108, 0x3393, 0x221a, 0x56a5, 0x472c, 0x75b7, 0x643e,
    0x9cc9, 0x8d40, 0xbfdb, 0xae52, 0xdaed, 0xcb64, 0xf9ff, 0xe876,
    0x2102, 0x308b, 0x0210, 0x1399, 0x6726, 0x76af, 0x4434, 0x55bd,
    0xad4a, 0xbcc3, 0x8e58, 0x9fd1, 0xeb6e, 0xfae7, 0xc87c, 0xd9f5,
    0x3183, 0x200a, 0x1291, 0x0318, 0x77a7, 0x662e, 0x54b5, 0x453c,
    0xbdcb, 0xac42, 0x9ed9, 0x8f50, 0xfbef, 0xea66, 0xd8fd, 0xc974,
    0x4204, 0x538d, 0x6116, 0x709f, 0x0420, 0x15a9, 0x2732, 0x36bb,
    0xce4c, 0xdfc5, 0xed5e, 0xfcd7, 0x8868, 0x99e1, 0xab7a, 0xbaf3,
    0x5285, 0x430c, 0x7197, 0x601e, 0x14a1, 0x0528, 0x37b3, 0x263a,
    0xdecd, 0xcf44, 0xfddf, 0xec56, 0x98e9, 0x8960, 0xbbfb, 0xaa72,
    0x6306, 0x728f, 0x4014, 0x519d, 0x2522, 0x34ab, 0x0630, 0x17b9,
    0xef4e, 0xfec7, 0xcc5c, 0xddd5, 0xa96a, 0xb8e3, 0x8a78, 0x9bf1,
    0x7387, 0x620e, 0x5095, 0x411c, 0x35a3, 0x242a, 0x16b1, 0x0738,
    0xffcf, 0xee46, 0xdcdd, 0xcd54, 0xb9eb, 0xa862, 0x9af9, 0x8b70,
    0x8408, 0x9581, 0xa71a, 0xb693, 0xc22c, 0xd3a5, 0xe13e, 0xf0b7,
    0x0840, 0x19c9, 0x2b52, 0x3adb, 0x4e64, 0x5fed, 0x6d76, 0x7cff,
    0x9489, 0x8500, 0xb79b, 0xa612, 0xd2ad, 0xc324, 0xf1bf, 0xe036,
    0x18c1, 0x0948, 0x3bd3, 0x2a5a, 0x5ee5, 0x4f6c, 0x7df7, 0x6c7e,
    0xa50a, 0xb483, 0x8618, 0x9791, 0xe32e, 0xf2a7, 0xc03c, 0xd1b5,
    0x2942, 0x38cb, 0x0a50, 0x1bd9, 0x6f66, 0x7eef, 0x4c74, 0x5dfd,
    0xb58b, 0xa402, 0x9699, 0x8710, 0xf3af, 0xe226, 0xd0bd, 0xc134,

```

```

0x39c3, 0x284a, 0x1ad1, 0x0b58, 0x7fe7, 0x6e6e, 0x5cf5, 0x4d7c,
0xc60c, 0xd785, 0xe51e, 0xf497, 0x8028, 0x91a1, 0xa33a, 0xb2b3,
0x4a44, 0x5bcd, 0x6956, 0x78df, 0x0c60, 0x1de9, 0x2f72, 0x3efb,
0xd68d, 0xc704, 0xf59f, 0xe416, 0x90a9, 0x8120, 0xb3bb, 0xa232,
0x5ac5, 0x4b4c, 0x79d7, 0x685e, 0x1ce1, 0x0d68, 0x3ff3, 0x2e7a,
0xe70e, 0xf687, 0xc41c, 0xd595, 0xa12a, 0xb0a3, 0x8238, 0x93b1,
0x6b46, 0x7acf, 0x4854, 0x59dd, 0x2d62, 0x3ceb, 0x0e70, 0x1ff9,
0xf78f, 0xe606, 0xd49d, 0xc514, 0xb1ab, 0xa022, 0x92b9, 0x8330,
0x7bc7, 0x6a4e, 0x58d5, 0x495c, 0x3de3, 0x2c6a, 0x1ef1, 0x0f78
};
#define PPPINITFCS16 0xffff /* Initial FCS value */
#define PPPGOODFCS16 0xf0b8 /* Good final FCS value */
/*
 * Calculate a new fcs given the current fcs and the new data.
 */
u16 pppfcs16(fcs, cp, len)
register u16 fcs;
register unsigned char *cp;
register int len;
{
    ASSERT(sizeof (u16) == 2);
    ASSERT(((u16) -1) > 0);
    while (len--)
        fcs=(fcs >> 8) ^ fcstab[(fcs ^ *cp++) & 0xff];
    return (fcs);
}
/*
 * How to use the fcs
 */
tryfcs16(cp, len)
register unsigned char *cp;
register int len;
{
    u16 trialfcs;
    /* add on output */
    trialfcs=pppfcs16( PPPINITFCS16, cp, len );
    trialfcs ^= 0xffff; /* complement */
    cp[len]=(trialfcs & 0x00ff); /* least significant byte first */
    cp[len+1]=((trialfcs >> 8) & 0x00ff);
    /* check on input */
    trialfcs=pppfcs16( PPPINITFCS16, cp, len + 2 );
    if ( trialfcs == PPPGOODFCS16 )

```

```

    printf("Good FCS\n");
}

```

D.2 校验表发生器

```

/*
 * Generate a FCS-16 table.
 * Drew D. Perkins at Carnegie Mellon University.
 * Code liberally borrowed from Mohsen Banan and D. Hugh Redelmeier.
 * The FCS-16 generator polynomial:  $x^{16} + x^{12} + x^5 + 1$ .
 */
#define P 0x8408
/*
 * NOTE The hex to "least significant bit" binary always causes
 * confusion, but it is used in all HDLC documents. Example: 03H
 * translates to 1100 0000B. The above defined polynomial value
 * (0x8408) is required by the algorithm to produce the results
 * corresponding to the given generator polynomial
 * ( $x^{16} + x^{12} + x^5 + 1$ )
 */
main()
{
    register unsigned int b, v;
    register int i;
    printf("typedef unsigned short u16;\n");
    printf("static u16 fcstab[256]={");
    for (b=0; ; )
    {
        if (b % 8 == 0)
            printf("\n");
        v=b;
        for (i=8; i--; )
            v=v & 1 ? (v >> 1) ^ P : v >> 1;
        printf("\t0x%04x", v & 0xFFFF);
        if (++b == 256)
            break;
        printf(", ");
    }
    printf("\n};\n");
}

```

D C

附 录 E
(规范性附录)
安全认证说明

E.1 比对加密方式

E.1.1 补位规则

待加密数据字节数不足待加密数据长度时，先补结束符0x80，剩余字节补0x00。比对因子、随机数和待加密数据均遵循此规则。

E.1.2 嵌有安全模块的电能表比对方案

嵌有安全模块的电能表比对方案如下：

- a) 提取比对因子：共 8 字节，从比对因子起始地址开始，在程序存储器中取 16 个字节。将 16 字节分为前 8 字节和后 8 字节两个数据块 (Data1, Data2)，然后对数据块进行处理 ($Data1 \hat{ } Data2 = Data3$ ，其中 $\hat{ }$ 代表异或运算符)，得到得到异或后数据块 Data3 作为比对因子；
- b) 提取随机数：共 16 字节，从比对数据起始地址开始取 64 字节，按长度平均分成四个数据块 (Data4, Data5, Data6, Data7)，然后对数据块进行处理 ($Data4 \hat{ } Data5 \hat{ } Data6 \hat{ } Data7 = Data8$ ，其中 $\hat{ }$ 代表异或运算符)，得到异或后数据块 Data8 作为随机数；
- c) 提取待加密数据：从比对数据起始地址开始取待加密数据长度字节，按长度平均分成四个数据块 (Data9, Data10, Data11, Data12)，然后对数据块进行处理 ($Data9 \hat{ } Data10 \hat{ } Data11 \hat{ } Data12 = Data13$ ，其中 $\hat{ }$ 代表异或运算符)，得到异或后数据块 Data13 (待加密数据长度必须为 64 整数倍，否则电能表返回异常应答“拒绝读写”)；
- d) 获取加密后数据：使用比对因子和随机数对数据块 Data13 进行加密，得到加密后数据。

E.1.3 未嵌安全模块的电能表比对方案

未嵌安全模块的电能表比对方案如下：

- a) 获取比对因子：从比对因子起始地址开始取待加密数据长度字节，按长度平均分成四个数据块 (Data1, Data2, Data3, Data4)，然后对数据块进行处理 ($Data1 \hat{ } Data2 \hat{ } Data3 \hat{ } Data4 = Data5$ ，其中 $\hat{ }$ 代表异或运算符)，得到比对因子 Data5；如果获取的比对因子中连续 16 字节为相同数据 (例全 00 或全 FF) 时，电能表应返回安全认证异常应答“地址异常”。
- b) 获取加密数据单元：从比对数据起始地址开始取待加密数据长度字节，按长度平均分成四个数据块 (Data6, Data7, Data8, Data9)，然后对数据块进行处理 ($Data6 \hat{ } Data7 \hat{ } Data8 \hat{ } Data9 = Data10$ ，其中 $\hat{ }$ 代表异或运算符)，得到加密数据单元 Data10。
- c) 获取代码密文：将比对因子 Data5 与加密数据单元 Data10 进行异或运算得到代码密文 Data11。

E.2 安全方式说明

说明如下：

- a) 如果每天收到的数据帧数据验证码校验失败、密文校验失败总累计达到 200 次，则终端远程设置参数、远程控制、清零功能挂起；在每日的零点，清除挂起状态及累计次数；
- b) 收到的抄读命令中数据验证码校验失败、密文校验失败不累计失败次数，挂起后能正常抄读。

E D

附 录 F
(资料性附录)
安全模式参数

F.1 安全模式参数

F.1.1 安全模式参数设置值的定义见表E.1。

表 E.1 安全模式参数设置值定义

编号	对象标识 OI	安全模式			
		读取	设置	操作	代理
字节数	2 字节	2 字节			

F.1.2 安全模式定义见表E.2。

表 E.2 安全模式定义

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
明文方式操作	明文+数据验证码操作	密文方式操作	密文+数据验证码操作	保留	代理读取	代理设置	代理操作
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
明文方式读取	明文+数据验证码读取	密文方式读取	密文+数据验证码读取	明文方式设置	明文+数据验证码设置	密文方式设置	密文+数据验证码设置

F.2 默认安全模式参数

默认安全模式参数的定义见表E.3。

表 E.3 默认安全模式参数

对象标识 OI	对象名称	读取			设置			操作			代理读取	代理设置	代理操作	
		明文	明文+数据验证码	密文	明文	明文+数据验证码	密文	明文	明文+数据验证码	密文				
0ZZZ	当前电能	√										√		

对象标识	对象名称	读取				设置				操作				代理读取	代理设置	代理操作
		明文	明文+数据验证码	密文	密文+数据验证码	明文	明文+数据验证码	密文	密文+数据验证码	明文	明文+数据验证码	密文	密文+数据验证码			
1ZZZ	最大需量		√													
202C	(当前)钱包文件	√											√			
2ZZZ	变量		√							√						
3ZZZ	事件		√					√				√				
4000	日期时间	√						√	√				√			
4001	通信地址	√						√					√			
4002	表号	√				√							√			
4003	客户编号		√			√										
400A	两套分时费率切换时间		√					√								
400B	两套阶梯切换时间		√					√								
401C	电流互感器变比		√					√								
401D	电压互感器变比		√					√								
401E	金额限值		√					√								
4018	当前套费率电价		√													
4019	备用套费率电价		√					√								
401A	当前套阶梯参数		√													
401B	备用套阶梯参数		√					√								
4111	备案号	√						√					√			
4ZZZ	参变量		√					√				√				
5000	瞬时冻结		√					√	√							
50ZZ	冻结		√					√				√				
60ZZ	采集	√				√			√				√	√	√	
70ZZ	集合	√							√				√		√	
80ZZ	控制		√					√				√				
F000	分帧传输	√						√				√				
F001	分块传输	√					√				√					
F002	扩展传输	√				√			√							
F100	ESAM	√				√						√	√			
F101	安全模式参数	√						√			√		√			
F2ZZ	输入输出接口设备	√				√			√				√	√	√	
FFZZ	自定义	√				√			√				√	√	√	

注1：Z代表本半字节所列数值的任意一个取值，但不能覆盖以上表格中已经列出的，例如以上表格中2ZZZ不能覆

盖202C、202E。

注2：其它数据如果在安全模式参数中没有明确要求，均采用明文+MAC方式读取。

注3：除以上表格中规定外，其它参数设置如果在安全模式参数中没有明确要求，均采用密文+MAC方式设置。

注4：默认安全模式参数和显式安全模式参数如果冲突，以显式安全模式参数为准。

注5：设置基表远程通信模块的信号强度时，不须硬件配合，不须密码验证。

F E

附录 G

(资料性附录)

状态字、特征字、模式字

G.1 电能表运行状态字 1

格式见表F.1。

表 F.1 电能表运行状态字 1

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
保留	计量芯片故障	无功功率方向 (0 正向、1 反向)	有功功率方向 (0 正向、1 反向)	停电抄表电池 (0 正常,1 欠压)	时钟电池 (0 正常,1 欠压)	需量积算方式 (0 滑差,1 区间)	保留
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
时钟故障	透支状态	存储器故障或损坏	内部程序错误	保留	保留	ESAM 错误	控制回路错误

G.2 电能表运行状态字 2

格式见表F.2。

表 F.2 电能表运行状态字 2

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
保留	C 相无功功率方向	B 相无功功率方向	A 相无功功率方向	保留	C 相有功功率方向	B 相有功功率方向	A 相有功功率方向
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
保留	保留	保留	保留	保留	保留	保留	保留
注：0代表正向，1代表反向。							

G.3 电能表运行状态字 3 (操作类)

格式见表F.3。

表 F.3 电能表运行状态字 3（操作类）

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
预跳闸报警状态 (0 无, 1 有)	继电器命令状态 (0 通, 1 断)	保留	继电器状态 (0 通, 1 断)	编程允许状态 (0 失效, 1 有效)	供电方式 (00 主电源, 01 辅助电源, 10 电池供电)		保留
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
远程开户 (0 开户, 1 未开户)	本地开户 (0 开户, 1 未开户)	安全认证状态 (0 失效, 1 有效)	保电状态 (0 非保电, 1 保电)	保留	保留	电能表类型 (00 非预付费表, 01 电量型预付费表, 10 电费型预付费表)	
<p>注 1: 编程允许状态(bit3): 对于有编程键的电能表, 此位为编程允许状态。</p> <p>注 2: 继电器状态(bit4), 指线路实际工作状态, 线路处于跳闸状态时此位置 1, 线路处于导通状态时此位置 0。</p> <p>注 3: 继电器远程拉闸命令状态(bit6)。电能表收到主站跳闸命令时, bit6 置 1; 电能表跳闸后, 该状态仍维持 1, 直到电能表解除跳闸条件, 或收到主站合闸、保电命令时将该位置 0。如果电能表处于保电状态时, 收到远程跳闸命令, 提示“拒绝操作”, 该位仍置 0。</p> <p>注 4: 预跳闸报警状态(bit7)是指剩余电量/金额小于等于预置的报警阈值 1 或电能表收到远程报警命令时, bit7 置 1, 电能表报警, 提示用户购电(或交费); 否则置 0。</p> <p>注 5: 电能表类型有非预付费型、电量型预付费和电费型预付费三种。当电能表类型为 00 时是非预付费型电能表(包括远程费控电能表); 当电能表类型为 01 时使用电量型预付费电能表; 当电能表类型为 10 时定义为电费型预付费电能表(包括本地费控电能表)。</p> <p>注 6: bit0、bit5、bit10、bit11 保留, 置 0。</p>							

G.4 电能表运行状态字 4（A相故障状态）

格式见表 F.4。

表 F.4 电能表运行状态字 4（A 相故障状态）

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
断相	功率反向	过载	过流	失流	过压	欠压	失压
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
保留	保留	保留	保留	保留	保留	保留	断流
注: 0代表无此类故障, 1代表当前发生此类故障。							

G.5 电能表运行状态字 5（B相故障状态）

格式见表 F.5。

表 F.5 电能表运行状态字 5 (B 相故障状态)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
断相	功率反向	过载	过流	失流	过压	欠压	失压
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
保留	保留	保留	保留	保留	保留	保留	断流
注：0代表无此类故障，1代表当前发生此类故障。							

G.6 电能表运行状态字 6 (C相故障状态)

格式见表F.6。

表 F.6 电能表运行状态字 6 (C 相故障状态)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
断相	功率反向	过载	过流	失流	过压	欠压	失压
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
保留	保留	保留	保留	保留	保留	保留	断流
注：0代表无此类故障，1代表当前发生此类故障。							

G.7 电能表运行状态字 7 (合相故障状态)

格式见表F.7。

表 F.7 电能表运行状态字 7 (合相故障状态)

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
总功率因数超下限	需量超限	掉电	辅助电源失电	电流不平衡	电压不平衡	电流逆相序	电压逆相序
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
保留	保留	保留	保留	保留	开端钮盖	开表盖	电流严重不平衡
注：0代表无此类故障，1代表当前发生此类故障。							

G.8 有功组合方式特征字

格式见表F.8。

表 F.8 有功组合方式特征字

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
保留	保留	保留	保留	反向有功 (0 不减, 1 减)	反向有功 (0 不加, 1 加)	正向有功 (0 不减, 1 减)	正向有功 (0 不加, 1 加)

G.9 无功组合方式 1、2 特征字

格式见表F.9 。

表 F.9 无功组合方式 1、2 特征字

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
IV 象限 (0 不减, 1 减)	IV 象限 (0 不加, 1 加)	III 象限 (0 不减, 1 减)	III 象限 (0 不加, 1 加)	II 象限 (0 不减, 1 减)	II 象限 (0 不加, 1 加)	I 象限 (0 不减, 1 减)	I 象限 (0 不加, 1 加)

G.10 周休日特征字

格式见表F.10 。

表 F.10 周休日特征字

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
保留	周六	周五	周四	周三	周二	周一	周日
注：0代表休息，1代表工作。							

G.11 插卡状态字

格式见表F.11 。

表 F.11 插卡状态字

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
保留	保留	保留	保留	保留	保留	插卡状态 (00 未知, 01 成功, 10 失败)	
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
保留	保留	保留	保留	保留	保留	保留	保留
注：bit0、bit1在插卡操作后置相应状态，读取和上电后置未知。							

G.12 控制命令执行状态字

格式见表F.12 。

表 F.12 控制命令执行状态字

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
保留	保留	延时跳闸 (大电流)	跳闸自动恢复	延时跳闸 (跳闸延时时间)	直接跳闸	允许合闸	直接合闸
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
保留	保留	保留	保留	报警解除	报警	保电解除	保电

G.13 控制命令错误状态字

格式见表F.13。

表 F.13 控制命令错误状态字

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
跳闸自动恢复时间无效	跳闸自动恢复命令执行失败(保电)	跳闸失败(保电)	保留	安全认证超时	密码错误/未授权	保留	电能表挂起
Bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
保留	保留	保留	保留	保留	保留	保留	跳闸自动恢复命令执行失败(跳闸)

注：表格未包含的其他错误，应答时均将bit2置1。

G.14 电能表跟随上报状态字

格式见表F.14。

表 F.14 电能表跟随上报状态字

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
保留	保留	存储器故障或损坏	保留	时钟电池欠压	保留	ESAM 错误	保留
bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
合闸成功	跳闸成功	保留	保留	保留	保留	透支状态	停电抄表电池欠压
bit23	bit22	bit21	bit20	bit19	bit18	bit17	bit16
保留	保留	管理模组升级	管理模组插拔	开端钮盖	开表盖	广播校时	普通校时
bit31	bit30	bit29	bit28	bit27	bit26	bit25	bit24
保留	保留	保留	保留	保留	保留	保留	保留

注1：电能表跟随上报模式字数据类型、数据格式同电能表跟随上报状态字，每位用于控制电能表跟随上报状态字中对应的位置位后是否上报。

G.15 计量模组系统状态字

格式见G.15。

表 F.15 计量芯系统状态字

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
时钟故障	保留	存储器故障或损坏	内部程序错误	时钟电池欠压	保留	ESAM 故障	保留
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
保留	保留	保留	保留	保留	保留	辅助电源失电	恒定磁场干扰

Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
保留	保留	管理模組升级	管理模組插拔	开端鈕盖	开表盖	广播校时	普通校时
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
保留	保留	保留	保留	保留	保留	保留	保留
<p>注 1：0 代表无此类故障，1 代表当前发生此类故障。</p> <p>注 2：时间故障、辅助电源失电、恒定磁场干扰由计量模組判断模式，管理模組根据状态事件记录。</p> <p>注 3：Bit16~31 为计量模組事件发生标志，事件一旦发生置对应的位，事件结束不清零对应的位，收到管理模組确认计量模組系统状态字命令才清零对应的位。确认计量模組系统状态字命令采用明文下发，确认命令参数中对应位如果为 1，则计量模組清零状态字中对应位，只能清零 Bit16~31。</p>							

G.16 采样状态字

格式见 G.16。

表 F.16 采样状态字

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
C 相无功功率 方向	B 相无功功率 方向	A 相无功功率 方向	总无功功率方 向	C 相有功功率 方向	B 相有功功率 方向	A 相有功功率 方向	总有功功率方 向
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
		电流逆相序	电压逆相序	C 相潜动	B 相潜动	A 相潜动	保留
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
保留	保留	保留	保留	00-57.7V, 01-100V, 02-220V		01-单相, 02-三相三线, 03-三相四线	
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
保留	保留	保留	保留	保留	保留	保留	保留
注：功率方向：0 代表正向，1 代表反向；潜动：0 代表处于起动，1 代表处于潜动。							

G F

附录 H
 (资料性附录)
 APDU 编码举例

H.1 预连接

H.1.1 登录

发送: 68 1E 00 81 05 07 09 19 05 16 20 00 HCS HCS 01 00 00 00 B4 07 E0 05 13 04 08 05 00 00 A4 FCS FCS 16 (斜体部分为链路层内容, 以下省略; 带下划线部分为APDU的A-XDR编码内容)

68 —— 帧起始符

1E 00 —— 长度域L=30字节

81 —— 控制域C

05 07 09 19 05 16 20 —— SA=05201605190907

00 —— CA

CS CS —— 帧头校验

01 —— [1] LINK-Request

00 —— PIID-ACD

00 —— 请求类型: 建立连接 (0)

00 B4 —— 心跳周期: 180s

07 E0 05 13 04 08 05 00 00 A4 —— 请求时间date_time: 2016-05-19 08: 05: 00: 0164

CS CS —— 帧校验

16 —— 结束符

响应: 68 30 00 01 05 07 09 19 05 16 20 10 CS CS 81 00 80 07 E0 05 13 04 08 05 00 00 89 07 E0 05 13 04 08 05 01 02 5F 07 E0 05 13 04 08 05 02 02 DA CS CS 16

68 —— 帧起始符

30 00 —— 长度域L=48字节

01 —— 控制域C

05 07 09 19 05 16 20 —— SA=05201605190907

10 —— CA

CS CS —— 帧头校验

81 —— [129] LINK-Response

00 —— PIID

80 —— 结果Result: 可信, 成功

07 E0 05 13 04 08 05 00 00 89 —— 请求时间date_time: 2016-05-19 08: 05: 00: 137

07 E0 05 13 04 08 05 01 02 5F —— 收到时间date_time: 2016-05-19 08: 05: 01: 607

07 E0 05 13 04 08 05 02 02 DA —— 响应时间date_time: 2016-05-19 08: 05: 02: 730

CS CS —— 帧校验

16 —— 结束符

H.1.2 心跳

心跳: 68 1E 00 81 05 07 09 19 05 16 20 00 CS CS 01 01 01 00 B4 07 E0 05 13 04 08 05 00
01 C3 CS CS 16

发送:

68 1E 00 81 05 07 09 19 05 16 20 00 CS CS

01 —— [1] LINK-Request

01 —— PIID--ACD

01 —— 请求类型: 心跳 (1)

00 B4 —— 心跳周期: 180s

07 E0 05 13 04 08 05 00 01 C3 —— 请求时间date_time: 2016-05-19 08: 05: 00: 451

CS CS 16

响应: 同建立连接的响应。

H.2 建立应用连接

发送: 02 00 00 10 FF
FF 04 00 04 00 01 04 00 00 00 00 64 00 00

02 —— [2] CONNECT-Request

00 —— PIID

00 10 —— 期望的协议版本号

FF FF FF FF FF FF FF FF —— ProtocolConformance

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF —— FunctionConformance

04 00 —— 客户机发送帧最大尺寸

04 00 —— 客户机接收帧最大尺寸

01 —— 客户机接收帧最大窗口尺寸

04 00 —— 客户机最大可处理APDU尺寸

00 00 00 64 —— 期望的应用连接超时时间

00 —— 认证请求对象 [0] NullSecurity

00 —— 没有时间标签

响应: 82 00 54 4F 50 53 30 31 30 32 31 36 30 37 33 31 30 31 30 32 31 36 30 37 33 31 00
00 00 00 00 00 00 00 00 10 FF
FF FF 04 00 04 00 01 04 00 00 00 00 64 00 00 00

82 —— [130] CONNECT-Response

00 —— PIID-ACD

54 4F 50 53 30 31 30 32 31 36 30 37 33 31 30 31 30 32 31 36 30 37 33 31 00 00 00 00
00 00 00 —— 厂商版本信息: 厂商代码 (size(4)) + 软件版本号 (size(4)) + 软件版本日期 (size(6))
+ 硬件版本号 (size(4)) + 硬件版本日期 (size(6)) + 厂商扩展信息 (size(8))

00 10 —— 期望的协议版本号

FF FF FF FF FF FF FF FF —— ProtocolConformance

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF —— FunctionConformance

04 00 —— 服务器发送帧最大尺寸

04 00 —— 服务器接收帧最大尺寸

01 —— 服务器接收帧最大窗口尺寸

04 00 —— 服务器最大可处理APDU尺寸
 00 00 00 64 —— 期望的应用连接超时时间
 00 —— 连接响应对象 允许建立应用连接 (0)
 00 —— 认证附加信息 OPTIONAL=0 表示没有
 00 —— FollowReport OPTIONAL=0表示没有上报信息
 00 —— 没有时间标签

H.3 读取

H.3.1 读取一个对象属性

读取电能表的通信地址，发送：05 01 01 40 01 02 00 00
 05 —— [5] GET-Request
 01 —— [1] GetRequestNormal
 01 —— PIID
 40 01 02 00 —— OAD：通信地址40010200
 00 —— 没有时间标签
 响应：85 01 01 40 01 02 00 01 09 06 12 34 56 78 90 12 00 00
 85 —— [133] GET-Response
 01 —— [1] GetResponseNormal
 01 —— PIID-ACD
 40 01 02 00 —— OAD
 01 —— Data
 09 —— octet-string
 06 —— SIZE(6)
 12 34 56 78 90 12 —— 通信地址：123456789012
 00 —— FollowReport OPTIONAL=0 表示没有上报信息
 00 —— 没有时间标签

H.3.2 读取多个对象属性

读取三相电压、电流，发送：05 02 02 02 20 00 02 00 20 01 02 00 00
 05 —— [5] GET-Request
 02 —— [2] GetRequestNormalList
 02 —— PIID
 02 —— SEQUENCE OF OAD，个数=2
 20 00 02 00 —— OAD1：A，B，C相计量电压
 20 01 02 00 —— OAD2：A，B，C相计量电流
 00 —— 没有时间标签
 响应：85 02 02 02 20 00 02 00 01 01 03 12 09 6D 12 09 6D 12 09 6D 20 01 02 00 01 01 03
 05 00 00 03 E8 05 00 00 03 E8 05 00 00 03 E8 00 00
 85 —— [133] GET-Response
 02 —— [2] GetResponseNormalList
 02 —— PIID-ACD

02 —— SEQUENCE OF A-ResultNormal, 个数=2
 20 00 02 00 —— OAD
 01 —— Data
 01 —— 类型=1, 表示数组
 03 —— 数组元素个数=3
 12 09 6D —— 元素1: 类型18: long-unsigned 241.3V A相
 12 09 6D —— 元素2: 类型18: long-unsigned 241.3V B相
 12 09 6D —— 元素3: 类型18: long-unsigned 241.3V C相
 20 01 02 00 —— OAD
 01 —— Data
 01 —— 类型=1, 表示数组
 03 —— 数组元素个数=3
 05 00 00 03 E8 —— 元素1: 类型: 5 double-long
 05 00 00 03 E8 —— 元素2: 类型: 5 double-long
 05 00 00 03 E8 —— 元素3: 类型: 5 double-long
 00 —— FollowReport OPTIONAL=0 表示没有上报信息
 00 —— 没有时间标签

H.3.3 读取一个记录型对象属性

(1) 终端读取电能表在2016-01-20 00: 00: 00的日冻结正向有功总及费率电能量

发送: 05 03 03 50 04 02 00 01 20 21 02 00 07 E0 16 01 14 00 00 00 02 00 20 21 02 00 00
 00 10 02 00 00
 05 —— [5] GET-Request
 03 —— [3] GetRequestRecord
 03 —— PIID
 50 04 02 00 —— OAD 日冻结
 01 —— RSD, 选择方法1
 20 21 02 00 —— OAD, 数据冻结时间
 1C 07 E0 01 14 00 00 00 —— 时间
 02 —— RCSD, SEQUENCE OF个数=2
 00 —— [0] OAD
 20 21 02 00 —— OAD
 00 —— [0] OAD
 00 10 02 00 —— OAD
 00 —— 没有时间标签
 响应: 85 03 03 50 04 02 00 02 00 20 21 02 00 00 00 10 02 00 01 01 1C 07 E0 01 14 00 00
 00 01 05 06 00 00 00 06 00 00 00 06 00 00 00 06 00 00 00 06 00 00 00 00 00 00
 85 —— [133] GET-Response
 03 —— [3] GeResponseRecord
 03 —— PIID-ACD
 50 04 02 00 —— OAD
 02 —— RCSD, SEQUENCE OF个数=2

00 20 21 02 00 —— 第1列OAD
 00 00 10 02 00 —— 第2列OAD
 01 —— 记录数据
 01 —— M条记录, M=1
 1C 07 E0 01 14 00 00 00 —— 第1列数据Data
 01 —— 第2列数据Data
 05
 06 00 00 00 00
 06 00 00 00 00
 06 00 00 00 00
 06 00 00 00 00
 06 00 00 00 00
 00 —— FollowReport OPTIONAL=0 表示没有上报信息
 00 —— 没有时间标签

(2) 主站召测集中器采集到的数据: 召测5个电能表的2016-01-20 00: 00: 00日冻结正向有功电能总及费率、日冻结反向有功电能总及费率

发送: 05 03 04 60 12 03 00 05 07 E0 01 14 00 00 00 03 05 06 04 00 00 00 01 21 06 04 00
 00 00 01 22 06 04 00 00 00 01 23 06 04 00 00 00 01 24 06 04 00 00 00 01 25 05 00 40 01 02 00
 00 60 40 02 00 00 60 41 02 00 00 60 42 02 00 01 50 04 02 00 02 00 10 02 00 00 20 02 00 00
 05 —— [5] GET-Request
 03 —— [3] GetRequestRecord
 04 —— PIID
 60 12 03 00 —— OAD
 05 —— RSD, 选择方法5
 07 E0 01 14 00 00 00 —— 采集存储时间
 03 —— 表计集合MS, 一组用户地址 [3] SEQUENCE OF TSA
 05 —— SEQUENCE OF TSA的个数=5
 06 04 00 00 00 01 21 —— TSA1
 06 04 00 00 00 01 22
 06 04 00 00 00 01 23
 06 04 00 00 00 01 24
 06 04 00 00 00 01 25 —— TSA5
 05 —— RCSD, SEQUENCE OF CSD个数=5
 00 40 01 02 00 —— OAD1, 通信地址
 00 60 40 02 00 —— OAD2, 采集启动时标
 00 60 41 02 00 —— OAD3, 采集成功时标
 00 60 42 02 00 —— OAD4, 采集存储时标
 01 —— ROAD5
 50 04 02 00 —— 日冻结
 02 —— 关联对象属性描述符 SEQUENCE OF个数=2
 00 10 02 00 —— 正向有功总及费率
 00 20 02 00 —— 反向有功总及费率

00 —— 没有时间标签

响应: 85 03 04 60 12 03 00 05 00 40 01 02 00 00 60 40 02 00 00 60 41 02 00 00 60 42 02
 00 01 50 04 02 00 02 00 10 02 00 00 20 02 00 01 05 09 05 00 00 00 01 21 1C 07 E0 01 14 00 00
 00 1C 07 E0 01 14 00 00 00 1C 07 E0 01 14 00 00 00 01 02 01 05 06 00 00 00 00 06 00 00 00 00
 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 01 05 06 00 00 00 00 06 00 00 00 00 06 00 00 00
 00 06 00 00 00 00 06 00 00 00 00 09 05 00 00 00 01 22 1C 07 E0 01 14 00 00 00 1C 07 E0 01 14
 00 00 00 1C 07 E0 01 14 00 00 00 01 02 01 05 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06
 00 00 00 00 06 00 00 00 00 01 05 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00
 06 00 00 00 00 09 05 00 00 00 01 23 1C 07 E0 01 14 00 00 00 1C 07 E0 01 14 00 00 00 1C 07 E0
 01 14 00 00 00 01 02 01 05 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00
 00 00 00 01 05 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 09
 05 00 00 00 01 24 1C 07 E0 01 14 00 00 00 1C 07 E0 01 14 00 00 00 1C 07 E0 01 14 00 00 00 01
 02 01 05 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 01 05 06
 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 09 05 00 00 00 01 25
 1C 07 E0 01 14 00 00 00 1C 07 E0 01 14 00 00 00 1C 07 E0 01 14 00 00 00 01 02 01 05 06 00 00
 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 01 05 06 00 00 00 06 00
 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 00 00 00 00 00 00 00

85 —— [133] GET-Response

03 —— [3] GeResponseRecord

04 —— PIID-ACD

60 12 03 00 —— OAD

05 —— RCSD, SEQUENCE OF个数=5, 即N=5

00 40 01 02 00 —— 第1列OAD

00 60 40 02 00 —— 第2列OAD

00 60 41 02 00 —— 第3列OAD

00 60 42 02 00 —— 第4列OAD

01 —— 第5列ROAD5

50 04 02 00 —— 日冻结

02

00 10 02 00 —— 正向有功总及费率

00 20 02 00 —— 反向有功总及费率

01 —— 记录数据

05 —— M条记录, M=5

09050000000121 1C07E00114000000 1C07E00114000000 1C07E00114000000 —— 第1条记录
 0102

01050600000000006000000000600000000060000000006000000000

01050600000000006000000000600000000060000000006000000000

09050000000122 1C07E00114000000 1C07E00114000000 1C07E00114000000 —— 第2条记录
 0102

01050600000000006000000000600000000060000000006000000000

01050600000000006000000000600000000060000000006000000000

09050000000123 1C07E00114000000 1C07E00114000000 1C07E00114000000 —— 第3条记录

0102

0105060000000006000000000600000000060000000006000000000

0105060000000006000000000600000000060000000006000000000

09050000000124 1C07E00114000000 1C07E00114000000 1C07E00114000000 —— 第4条记录

0102

0105060000000006000000000600000000060000000006000000000

0105060000000006000000000600000000060000000006000000000

09050000000125 1C07E00114000000 1C07E00114000000 1C07E00114000000 —— 第5条记录

0102

0105060000000006000000000600000000060000000006000000000

0105060000000006000000000600000000060000000006000000000

00 —— FollowReport OPTIONAL=0 表示没有上报信息

00 —— 没有时间标签

H.3.4 读取多个记录型对象属性

参考H.3.3。

H.3.5 读取分帧响应的下一个数据块

读取全部电能表档案，发送：05 01 08 60 00 02 00 00

05 —— [5] GET-Request

01 —— [1] GetRequestNormal

08 —— PIID

60 00 02 00 —— OAD

00 —— 没有时间标签

响应：85 05 08 00 00 01 01 01 60 00 02 00 01 01 03 02 04 12 00 01 02 0A 55 06 04 00 00

00 22 21 16 03 16 03 51 F2 09 02 01 09 06 00 00 00 00 00 00 00 11 04 11 04 16 01 12 08 98 12 00

0F 02 04 55 06 00 00 00 00 00 00 00 09 06 00 00 00 00 00 00 12 00 01 12 00 01 01 00 02 04 12 00

02 02 0A 55 06 04 00 00 00 22 22 16 03 16 03 51 F2 09 02 01 09 06 00 00 00 00 00 00 11 04 11

04 16 01 12 08 98 12 00 0F 02 04 55 06 00 00 00 00 00 09 06 00 00 00 00 00 12 00 01 12

00 01 01 00 02 04 12 00 03 02 0A 55 06 04 00 00 00 22 23 16 03 16 03 51 F2 09 02 01 09 06 00

00 00 00 00 00 11 04 11 04 16 01 12 08 98 12 00 0F 02 04 55 06 00 00 00 00 00 09 06 00 00

00 00 00 00 12 00 01 12 00 01 01 00 00 00

85 —— [133] GET-Response

05 —— [5] GetResponseNext

08 —— PIID-ACD

00 —— 末帧标志 False

00 01 —— 分帧序号=1

01 —— 分帧响应，对象属性 [1] SEQUENCE OF A-ResultNormal

01 —— SEQUENCE OF个数=1

60 00 02 00 —— OAD

01 —— data

01 —— array

03 —— 个数=3

02 04 12 00 01 02 0A 55 06 04 00 00 00 22 21 16 03 16 03 51 F2 09 02 01 09 06 00 00 00
00 00 00 11 04 11 04 16 01 12 08 98 12 00 0F 02 04 55 06 00 00 00 00 00 00 09 06 00 00 00 00
00 00 12 00 01 12 00 01 01 00

02 04 12 00 02 02 0A 55 06 04 00 00 00 22 22 16 03 16 03 51 F2 09 02 01 09 06 00 00 00
00 00 00 11 04 11 04 16 01 12 08 98 12 00 0F 02 04 55 06 00 00 00 00 00 00 09 06 00 00 00 00
00 00 12 00 01 12 00 01 01 00

02 04 12 00 03 02 0A 55 06 04 00 00 00 22 23 16 03 16 03 51 F2 09 02 01 09 06 00 00 00
00 00 00 11 04 11 04 16 01 12 08 98 12 00 0F 02 04 55 06 00 00 00 00 00 00 09 06 00 00 00 00
00 00 12 00 01 12 00 01 01 00

00 —— FollowReport OPTIONAL=0 表示没有上报信息

00 —— 没有时间标签

读分帧的下一帧：

发送：05 05 09 00 01 00

05 —— [5] GET-Request

05 —— [5] GetRequestNext

09 —— PIID

00 01 —— 正确接收的最近一次数据块序号

00 —— 没有时间标签

响应：85 05 09 01 00 02 01 01 60 00 02 00 01 01 03 02 04 12 00 04 02 0A 55 06 04 00 00
00 22 24 16 03 16 03 51 F2 09 02 01 09 06 00 00 00 00 00 00 11 04 11 04 16 01 12 08 98 12 00
0F 02 04 55 06 00 00 00 00 00 00 09 06 00 00 00 00 00 00 12 00 01 12 00 01 01 00 02 04 12 00
05 02 0A 55 06 04 00 00 00 22 25 16 03 16 03 51 F2 09 02 01 09 06 00 00 00 00 00 00 11 04 11
04 16 01 12 08 98 12 00 0F 02 04 55 06 00 00 00 00 00 00 09 06 00 00 00 00 00 12 00 01 12
00 01 01 00 02 04 12 00 06 02 0A 55 06 04 00 00 00 22 26 16 03 16 03 51 F2 09 02 01 09 06 00
00 00 00 00 00 11 04 11 04 16 01 12 08 98 12 00 0F 02 04 55 06 00 00 00 00 00 09 06 00 00
00 00 00 00 12 00 01 12 00 01 01 00 00 00

85 —— [133] GET-Response

05 —— [5] GeResponseNext

09 —— PIID-ACD

01 —— 末帧标志 True

00 02 —— 分帧序号=2

01 —— 分帧响应，对象属性 [1] SEQUENCE OF A-ResultNormal

01 —— SEQUENCE OF个数=1

60 00 02 00 —— OAD

01 —— data

01 —— Data

03 —— 个数=3

02 04 12 00 04 02 0A 55 06 04 00 00 00 22 24 16 03 16 03 51 F2 09 02 01 09 06 00 00 00
00 00 00 11 04 11 04 16 01 12 08 98 12 00 0F 02 04 55 06 00 00 00 00 00 00 09 06 00 00 00 00
00 00 12 00 01 12 00 01 01 00 —— 第1个数据

02 04 12 00 05 02 0A 55 06 04 00 00 00 22 25 16 03 16 03 51 F2 09 02 01 09 06 00 00 00
 00 00 00 11 04 11 04 16 01 12 08 98 12 00 0F 02 04 55 06 00 00 00 00 00 09 06 00 00 00 00
 00 00 12 00 01 12 00 01 01 00 —— 第2个数据

02 04 12 00 06 02 0A 55 06 04 00 00 00 22 26 16 03 16 03 51 F2 09 02 01 09 06 00 00 00
 00 00 00 11 04 11 04 16 01 12 08 98 12 00 0F 02 04 55 06 00 00 00 00 00 09 06 00 00 00 00
 00 00 12 00 01 12 00 01 01 00 —— 第3个数据

00 —— FollowReport OPTIONAL=0 表示没有上报信息

00 —— 没有时间标签

H.3.6 读取一个对象属性（明文+MAC方式）

读取电能表的通信地址，发送：68 L L 43 05 01 00 29 01 16 20 0A HCS_L HCS_H 10 00 08 05 01
 01 40 01 02 00 00 00 85 01 02 03 06 12 34 56 78 90 12 04 12 34 56 78 FCS_L FCS_H 16

10 —— [16] SECURITY-Request

00 —— 明文

08 —— 明文的长度

05 —— [5] GET-Request

01 —— [1] GetRequestNormal

01 —— PIID

40 01 02 00 —— OAD: 通信地址40010200

00 —— 没有时间标签

00 —— 数据验证信息 [0] SID_MAC

85 01 02 03 —— 标识

06 12 34 56 78 90 12 —— 附加数据

04 —— MAC 的长度

12 34 56 78 —— MAC

响应：68 L L C3 05 01 00 29 01 16 20 0A HCS_L HCS_H 90 00 11 85 01 01 40 01 02 00 01 09
 06 20 16 01 29 00 01 00 01 00 04 12 34 56 78 FCS_L FCS_H 16

90 —— [144] SECURITY-Response

00 —— 明文

12 —— 明文的长度

85 —— [133] GET-Response

01 —— [1] GetResponseNormal

01 —— PIID-ACD

40 01 02 00 —— OAD

01 —— Data

09 —— octet-string

06 —— SIZE(6)

20 16 01 29 00 01 —— 通信地址：201601290001

00 —— 没有跟随上报信息

00 —— 没有时间标签

01 —— 含数据验证信息

00 —— 数据验证信息 [0] MAC

04 —— MAC 的长度
12 34 56 78 —— MAC

H.4 设置

H.4.1 设置一个对象属性请求

设置时钟，发送：06 01 02 40 00 02 00 1C 07 E0 01 14 10 1B 0B 00
06 —— [6] SET-Request
01 —— [1] SetRequestNormal
02 —— PIID
40 00 02 00 —— OAD
1C —— Data: 类型28: date_time_s
07 E0 01 14 10 1B 0B —— 时间: 2016-01-20 16: 27: 11
00 —— 没有时间标签
响应：86 01 02 40 00 02 00 00 00 00
86 —— [134] SET-Response
01 —— [1] SetResponseNormal
02 —— PIID-ACD
40 00 02 00 —— OAD
00 —— DAR, 0成功
00 —— FollowReport OPTIONAL=0 表示没有上报信息
00 —— 没有时间标签

H.4.2 设置多个对象属性请求

设置通信地址和时钟，发送：06 02 03 02 40 01 02 00 09 06 00 00 00 00 01 40 00 02 00 1C
07 E0 01 14 10 1B 0B 00
06 —— [6] SET-Request
02 —— [2] SetRequestNormalList
03 —— PIID
02 —— SEQUENCE OF的个数=2
40 01 02 00 —— OAD1
09 06 00 00 00 00 00 01 —— value1 (通信地址值: 000000000001)
40 00 02 00 —— OAD2
1C 07 E0 01 14 10 1B 0B —— value2 (时间: 2016-01-20 16:27:11)
00 —— 没有时间标签
响应：86 02 03 02 40 01 02 00 00 40 00 02 00 00 00 00
86 —— [134] SET-Response
02 —— [2] SetResponseNormalList
03 —— PIID-ACD
02 —— SEQUENCE OF个数=2
40 01 02 00 —— OAD1
00 —— DAR1, 0成功

40 00 02 00 —— OAD2
 00 —— DAR2, 0成功
 00 —— FollowReport OPTIONAL=0 表示没有上报信息
 00 —— 没有时间标签

H.4.3 设置后读取多个对象属性请求

设置心跳周期后读取, 发送: 06 03 04 01 45 00 02 0C 12 01 68 45 00 02 0C 03 00
 06 —— [6] SET-Request
 03 —— [3] SetThenGetRequestNormalList
 04 —— PIID
 01 —— SEQUENCE OF的个数=1
 45 00 02 0C —— set OAD: 心跳周期
 12 01 68 —— Data
 45 00 02 0C —— read OAD: 心跳周期
 03 —— 读取延时 3秒
 00 —— 没有时间标签
 响应: 86 03 04 01 45 00 02 0C 00 45 00 02 0C 01 12 01 68 00 00
 86 —— [134] SET-Response
 03 —— [3] SetThenGetResponseNormalList
 04 —— PIID-ACD
 01 —— SEQUENCE OF的个数=1
 45 00 02 0C —— set OAD: 心跳周期
 00 —— DAR, 0成功
 45 00 02 0C —— read OAD: 心跳周期
 01 —— Data flag
 12 01 68 —— Data
 00 —— FollowReport OPTIONAL=0 表示没有上报信息
 00 —— 没有时间标签

H.5 操作

H.5.1 操作一个对象方法请求

执行电能量复位方法, 发送: 07 01 05 00 10 01 00 0F 00 00
 07 —— [7] ACTION-Request
 01 —— [1] ActionRequest
 05 —— PIID
 00 10 01 00 —— OMD
 0F 00 —— 参数Data, integer(0)
 00 —— 没有时间标签
 响应: 87 01 05 00 10 01 00 00 00 00
 87 —— [135] ACTION-Response
 01 —— [1] ActionResponseNormal

H.5.4 添加普通采集方案

(1) 采集当前数据

发送：07 01 05 60 14 7F 00 01 01 02 06 11 01 12 00 01 02 02 11 00 00 01 04 5B 00 00 10
02 00 5B 00 00 20 02 00 5B 00 20 00 02 00 5B 00 20 01 02 00 5C 01 16 01 00
07 —— [7] ACTION-Request
01 —— [1] ActionRequest
05 —— PIID
60 14 7F 00 —— 普通采集方案 方法127
01 —— ARRAY
01 —— 1组普通采集方案
02 —— 结构体
06 —— 成员数量
11 01 —— 方案编号 1
12 00 01 —— 存储深度 1
02 —— 结构体
02 —— 成员数量
11 00 —— 方案类型 0
00 —— 采集内容 NULL 采集当前数据
01 —— ARRAY
04 —— CSD数量为4
5B 00 00 10 02 00 —— OAD 00100200
5B 00 00 20 02 00 —— OAD 00200200
5B 00 20 00 02 00 —— OAD 20000200
5B 00 20 01 02 00 —— OAD 20010200
5C —— 表计集合数据类型
01 —— 表计集合：全部用户地址
16 —— enum类型
01 —— 存储时标选择：任务开始时间
00 —— 没有时间标签
响应：87 01 05 60 14 7F 00 00 00 00 00
87 —— [135] ACTION-Response
01 —— [1] ActionResponseNormal
05 —— PIID-ACD
60 14 7F 00 —— OMD
00 —— DAR, 0成功
00 —— Data OPTIONAL=0 表示没有数据
00 —— FollowReport OPTIONAL=0 表示没有上报信息
00 —— 没有时间标签

(2) 采集冻结数据

发送：07 01 05 60 14 7F 00 01 01 02 06 11 01 12 00 20 02 02 11 02 00 01 01 5B 01 50 04
02 00 04 00 10 02 00 00 20 02 00 00 30 02 00 00 40 02 00 5C 01 16 02 00
07 —— [7] ACTION-Request

01 —— [1] ActionRequest
 05 —— PIID
 60 14 7F 00 —— 普通采集方案 方法127
 01 —— ARRAY
 01 —— 1组普通采集方案
 02 —— 结构体
 06 —— 成员数量
 11 01 —— 方案编号 1
 12 00 20 —— 存储深度 32
 02 —— 结构体
 02 —— 成员数量
 11 02 —— 方案类型 2
 00 —— 采集内容 NULL 按冻结时标采集
 01 —— ARRAY
 01 —— CSD数量为1
 5B 01 —— ROAD
 50 04 02 00 —— OAD
 04 —— 关联对象个数
 00 10 02 00 —— OAD 00100200
 00 20 02 00 —— OAD 00200200
 00 30 02 00 —— OAD 00300200
 00 40 02 00 —— OAD 00400200
 5C —— 表计集合数据类型
 01 —— 表计集合：全部用户地址
 16 —— enum类型
 02 —— 存储时标选择：相对当日0点0分
 00 —— 没有时间标签
 响应：87 01 05 60 14 7F 00 00 00 00 00
 87 —— [135] ACTION-Response
 01 —— [1] ActionResponseNormal
 05 —— PIID-ACD
 60 14 7F 00 —— OMD
 00 —— DAR, 0成功
 00 —— Data OPTIONAL=0 表示没有数据
 00 —— FollowReport OPTIONAL=0 表示没有上报信息
 00 —— 没有时间标签

H.6 上报

H.6.1 上报多个记录型对象属性

参考H.3.4。

H.7 代理

H.7.1 代理读取多个服务器的多个对象属性

代理读取2个电能表的当前电能量，发送：09 01 0A 00 78 02 07 05 20 16 01 20 00 01 00 3C 01
00 10 02 00 07 05 20 16 01 20 00 02 00 3C 01 00 10 02 00 00

09 —— [9] PROXY-Request
01 —— [1] ProxyGetRequestList
0A —— PIID
00 78 —— 整个代理请求的超时时间
02 —— SEQUENCE OF个数=2
07 05 20 16 01 20 00 01 —— TSA
00 3C —— 代理一个服务器的超时时间
01 —— SEQUENCE OF OAD个数=1
00 10 02 00 —— OAD
07 05 20 16 01 20 00 02 —— TSA
00 3C —— 代理一个服务器的超时时间
01 —— SEQUENCE OF个数=1
00 10 02 00 —— OAD
00 —— 没有时间标签

响应：89 01 0A 02 07 05 20 16 01 20 00 01 01 00 10 02 00 01 01 05 06 00 00 00 00 06 00
00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 07 05 20 16 01 20 00 02 01 00 10 02 00
01 01 05 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 00

89 —— [137] PROXY-Response
01 —— [1] ProxyGetResponseList
0A —— PIID-ACD
02 —— SEQUENCE OF个数=2
07 05 20 16 01 20 00 01 —— TSA
01 —— SEQUENCE OF个数=1
00 10 02 00 —— OAD
01 —— Data
01 —— array
05 —— 5个元素
06 00 00 00 00 —— 总
06 00 00 00 00 —— 费率1
06 00 00 00 00 —— 费率2
06 00 00 00 00 —— 费率3
06 00 00 00 00 —— 费率4
07 05 20 16 01 20 00 02 —— TSA
01 —— SEQUENCE OF个数=1
00 10 02 00 —— OAD
01 —— Data
01 —— array

05 —— 5个元素
06 00 00 00 00 —— 总
06 00 00 00 00 —— 费率1
06 00 00 00 00 —— 费率2
06 00 00 00 00 —— 费率3
06 00 00 00 00 —— 费率4
00 —— FollowReport OPTIONAL=0 表示没有上报信息
00 —— 没有时间标签

H. 7. 2 代理读取一个服务器的一个记录型对象属性

参考H. 3. 3 。

H. 7. 3 代理设置多个服务器的多个对象属性

参考H. 4. 2 。

H. 7. 4 代理设置后读取多个服务器的多个对象属性

参考H. 4. 3 。

H. 7. 5 代理操作多个服务器的多个对象方法

参考H. 5. 2 。

H. 7. 6 代理操作后读取多个服务器的多个对象方法和属性

参考H. 5. 3 。

面向对象的用电信息数据交换协议

编 制 说 明

目 次

1 编制背景.....	237
2 编制主要原则.....	237
3 与其他标准文件的关系.....	237
4 主要工作过程.....	237
5 标准结构和内容.....	238
6 条文说明.....	238

1 编制背景

本标准依据《国家电网公司关于下达 2017 年度公司第一批技术标准制修订计划的通知》（国家电网科〔2017〕72 号）的要求编写。

随着用电信息采集领域业务需求的快速发展，多功能电能表通信协议（DL/T 645—2007）因各种扩展而很不统一，使得用电信息采集通信过程中增加了许多协议转换问题，难以实现互操作，严重制约了用电信息采集系统的应用与发展。目前国网公司企业标准 Q/GDW 1376.1—2013 远程通信协议是面向业务数据的协议，对于用电信息采集中日趋多样化的采集任务需求，其在拓展性、复用性和灵活性方面逐渐凸现不足。

本标准编制的主要目的是提出一种采用面向对象思想建模的，高度灵活、可配置和可扩展的用电信息数据交换协议，能够满足业务需求变化，以提高用电信息采集成功率、业务执行成功率、数据溯源性和安全性。

2 编制主要原则

本标准参考现有的行业标准、国家标准及国际标准，遵循先进性与实用性相结合、统一性与灵活性相结合、可靠性与经济性相结合的原则，规定面向对象的用电信息数据交换协议。

3 与其他标准文件的关系

本标准与相关技术领域的国家现行法律、法规和政策保持一致。
本标准不涉及专利、软件著作权等知识产权使用问题。

4 主要工作过程

2017 年 1 月至 2 月，调研有关用电信息采集系统通信协议的应用情况，收集整理国内外相关技术标准，编制《面向对象的用电信息数据交换协议》标准工作方案。

2017 年 2 月 10 日，中国电力科学研究院在北京组织召开第一次《面向对象的用电信息数据交换协议》技术标准编制启动工作会议，审议《面向对象的用电信息数据交换协议》技术标准编制工作方案，组建标准编制工作组，落实标准编制内容和任务分工，正式启动标准编制工作。标准编制工作组由中国电力科学研究院、国网浙江电力、国网湖南电力、国网冀北电力、国网福建电力、国网安徽电力等单位组成。

2017 年 2 月，按照标准编制工作方案的要求，分组编写标准内容，由小组组长牵头，按时完成技术标准各部分内容的编制工作，形成技术标准工作组讨论稿。

2017 年 3 月，形成征求意见稿，在国家电网公司范围内开展技术标准征求意见。

2017 年 3 月 21 日，标准编制工作组在北京召开第二次《面向对象的用电信息数据交换协议》技术标准编制研讨会。标准编制工作组成员对征求意见稿内容进行讨论，对技术标准进行了完善。

2017 年 3 月至 4 月，在湖南、浙江等开展现场试点验证，重点验证了面向对象的用电信息数据交换协议的采集效率、业务执行成功率，以及对全事件采集、多表集抄等业务的支撑能力。

2017 年 4 月 26 日，标准编制工作组在在湖南召开面向对象的用电信息数据交换协议试点应用现场技术研讨会，结合试点应用情况对技术标准进行完善，形成技术标准送讨论审稿。

2017 年 5 月 30 日，标准编制工作组在北京召开第三次《面向对象的用电信息数据交换协议》技术

标准编制研讨会，结合试点应用情况对技术标准进行了完善，形成了技术标准送审稿。

2017年6月8日，在北京召开标准审查会，通过了标准审查组审查，标准编制组根据审查意见对技术标准进行了完善，形成了技术标准报批稿。

5 标准结构和内容

本标准依据《国家电网公司技术标准管理办法》（国家电网企管〔2014〕455号文）的要求编制。

本标准的主要结构和内容如下：

本标准主题章为4章，由通信架构、数据链路层、应用层、接口类与对象标识组成。通信架构叙述了本协议客户机/服务器的通信模型、面向应用连接的数据交换及数据交换类型，还给出了服务器的基本模型。数据链路层规定本协议链路层采用的线性帧结构，具有结构清晰、编码效率高等特点。应用层规定了通信协议规范、以及应用层数据单元（APDU）的数据类型规范和遵循的APDU编码规范。接口类与对象标识，描述了对象模型，规定了接口类描述方法，定义了用电信息采集相关的17个对象接口类，规定了对象标识系统。

6 条文说明

本标准第1章中，当主站与采集终端或主站与电能表通信时，采集终端和电能表统一简称为“终端”。
