

ICS 29.240

F 20

备案号: 61627-2018



# 中华人民共和国电力行业标准

DL / T 698.45 — 2017

代替 DL / T 698 — 1999

## 电能信息采集与管理系统 第 4-5 部分：通信协议——面向 对象的数据交换协议

Data acquisition and management system for electrical energy  
Part 4-5: Communication protocol - object oriented data exchange protocol

2017-11-15 发布

2018-03-01 实施

国家能源局 发布

## 目 次

前言	II
1 范围	1
2 规范性引用文件	1
3 术语、定义及符号和缩略语	1
3.1 术语和定义	1
3.2 符号和缩略语	2
4 通信架构	3
4.1 信息交换模型	3
4.2 应用连接的数据交换	4
4.3 请求/响应类型的数据交换	4
4.4 通知/确认类型的数据交换	4
4.5 服务器模型	5
5 数据链路层	5
5.1 帧结构	5
5.2 字节格式	9
5.3 传输规则	9
6 应用层	11
6.1 应用层服务规范	11
6.2 应用层协议规范	23
6.3 应用层数据单元规范	27
6.4 应用层编码规范	63
7 接口类与对象标识	63
7.1 对象模型	63
7.2 接口类的表示方法	63
7.3 接口类	64
7.4 对象标识	103
附录 A (资料性附录) 校验算法	115
附录 B (规范性附录) 物理单位枚举	118
附录 C (规范性附录) 有关一致性协商	121
附录 D (资料性附录) APDU 编码举例	123
附录 E (规范性附录) 对象标识定义	139
附录 F (资料性附录) 状态字、特征字、模式字	196
附录 G (规范性附录) 安全认证说明	201
附录 H (资料性附录) 安全模式参数	202



## 前 言

本部分依据 GB/T 1.1—2009《标准化工作导则 第1部分：标准的结构和编写》给出的规则起草。

DL/T 698 电能信息采集与管理系统分为以下部分：

DL/T 698.1 电能信息采集与管理系统 第1部分：导则；

DL/T 698.2 电能信息采集与管理系统 第2部分：主站技术规范；

DL/T 698.31 电能信息采集与管理系统 第3-1部分：电能信息采集终端技术规范——通用要求；

DL/T 698.32 电能信息采集与管理系统 第3-2部分：电能信息采集终端技术规范——厂站采集终端特殊要求；

DL/T 698.33 电能信息采集与管理系统 第3-3部分：电能信息采集终端技术规范——专变采集终端特殊要求；

DL/T 698.34 电能信息采集与管理系统 第3-4部分：电能信息采集终端技术规范——公变采集终端特殊要求；

DL/T 698.35 电能信息采集与管理系统 第3-5部分：电能信息采集终端技术规范——低压集中抄表终端特殊要求；

DL/T 698.36 电能信息采集与管理系统 第3-6部分：电能信息采集终端技术规范——通信单元要求；

DL/T 698.41 电能信息采集与管理系统 第4-1部分：通信协议——主站与电能信息采集终端通信；

DL/T 698.42 电能信息采集与管理系统 第4-2部分：通信协议——集中器本地通信接口协议；

DL/T 698.44 电能信息采集与管理系统 第4-4部分：通信协议——微功率无线通信；

DL/T 698.45 电能信息采集与管理系统 第4-5部分：通信协议——面向对象的数据交换协议；

DL/T 698.46 电能信息采集与管理系统 第4-6部分：通信协议——采集终端远程通信模块接口协议；

DL/T 698.51 电能信息采集与管理系统 第5-1部分：测试技术规范——功能测试；

DL/T 698.52 电能信息采集与管理系统 第5-2部分：测试技术规范——远程通信协议一致性测试。

本部分为 DL/T 698 的第4-5部分，为新增部分。

本部分由中国电力企业联合会提出。

本部分由电力行业电测量标准化技术委员会（DL/TC22）归口。

本部分起草单位：中国电力科学研究院有限公司、国家电网有限公司、国网浙江省电力有限公司、国网湖南省电力有限公司、国网冀北电力有限公司、国网福建省电力有限公司、国网江苏省电力有限公司、青岛鼎信通讯股份有限公司、江苏林洋能源服务有限公司、上海协同科技股份有限公司、长沙威胜信息技术有限公司、河南许继仪表有限公司、深圳市国电科技通信有限公司、南京新联电子股份有限公司、杭州海兴电力科技股份有限公司、宁波三星医疗电气股份有限公司、杭州炬华科技股份有限公司、烟台东方威思顿电气有限公司、北京智芯微电子科技有限公司。

本部分主要起草人：郑安刚、巫钟兴、刘宣、阿辽沙·叶、祝恩国、郜波、姜洪浪、翟峰、付义伦、杜新纲、葛得辉、彭楚宁、周晖、章欣、徐英辉、赵兵、孟静、邹和平、刘兴奇、赵婷、王伟峰、郑松松、王朝亮、朱佳柯、陈石东、巨汉基、夏桃芳、徐晴、周超、范建华、孟宇、蒋长献、李力、李耀、谭赣江、刘永波、顾君、张小秋、冯秋雄、陈杰、陈飞虎、魏灵坤、王于波。

本部分在执行过程中的意见或建议反馈至中国电力企业联合会标准化管理中心（北京市白广路二条一号，100761）。

## 电能信息采集与管理系统

### 第 4-5 部分：通信协议——面向对象的数据交换协议

#### 1 范围

DL/T 698 的本部分规定了电能信息采集与管理系统主站（以下简称“主站”）、采集终端或电能表（以下简称“终端”）之间，采用的面向对象具有互操作性的数据传输协议，包括通信架构、数据链路层、应用层、以及接口类及其对象和对象标识。

本部分适用于主站、采集终端、电能表之间的通信数据交换。水、气、热等表计的信息采集可参照使用。

#### 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

#### 3 术语、定义及符号和缩略语

##### 3.1 术语和定义

下列术语和定义适用于本文件。

##### 3.1.1

**面向对象的数据交换协议** **object oriented data exchange protocol**

基于面向对象建模方法建立的一套适用于采集系统的互操作性数据交换通信协议。

注：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。

##### 3.1.2

**对象标识** **object identification**

用于标识终端中对象唯一名称的编码。

##### 3.1.3

**逻辑名** **logical name**

用于标识接口类的实例。

注：逻辑名是接口类的第一个属性，它的值与对象标识一致。

##### 3.1.4

**类标识码** **class id**

用于区别对象接口类的标识码，相当于接口类的名称。

##### 3.1.5

**服务器地址** **server address**

指客户机/服务器（client/server）访问模型中的服务器（server）的通信地址。

3.1.6

**逻辑地址 logic address**

指终端的服务器模型中逻辑设备的地址。

3.1.7

**客户机地址 client address**

指客户机/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。

3.1.8

**采集启动时标 acquisition start time**

指启动一次采集任务时的设备时钟当前值。

注：采集启动时标值只与启动时刻有关，与执行的时间长短无关。

3.1.9

**采集成功时标 acquisition time**

指客户机成功接收到服务器响应时的设备时钟当前值。

3.1.10

**采集存储时标 acquisition storage time**

采集到的数据进行存储的时间。

3.1.11

**组地址 group address**

具有某一相同属性的设备群组编码。

注：如组地址属于同一行业、同一变电站、同一线路，可以响应同一个命令。

3.1.12

**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。

3.1.13

**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。

3.2 符号和缩略语

符号和缩略语见表 1。

表 1 符号和缩略语

符号和缩略语	全 称	含 义
A	address	地址域
ACD	ask call demand	请求访问标识
AD	acquired data	采集数据
APDU	application layer data unit	应用层数据单元
A-XDR	adapted extended data representation	可调整的扩展数据表示
B	binary	二进制
CA	client address	客户机通信地址

表 1（续）

符号和缩略语	全 称	含 义
CSD	column selection descriptor	列选择描述符
DAR	data access result	数据访问结果
DIR	direction	传输方向位
ESAM	embedded secure access module	嵌入式安全控制模块
FCS	frame check sum	帧校验
H	hex	十六进制
HCS	head check sum	帧头校验
IC	interface class	接口类
LSB	least significant bit	最低有效位
MAC	message authentication code	消息鉴别码
MS	meter set	表计集合
OAD	object attribute descriptor	对象属性描述符
OI	object identify	对象标识
OMD	object method descriptor	对象方法描述符
PIID	priority and invoke ID	序号及优先标识
PIID-ACD	priority and invoke ID with ACD	带请求访问标识的序号及优先标识
PRM	primary request message	启动标识位
RCSD	record column selection descriptor	记录列选择描述符
RN	random numbers	随机数
ROAD	record object attribute descriptor	记录型对象属性描述符
RSD	record selection descriptor	记录选择描述符
SA	server address	服务器地址
SC	scrambling code	扰码
TI	time interval	时间间隔
TSA	target server address	目标服务器地址
::=		定义为

## 4 通信架构

### 4.1 信息交换模型

本部分信息交换模型见图 1。客户机和服务器之间的信息交换借助于通信协议实现。

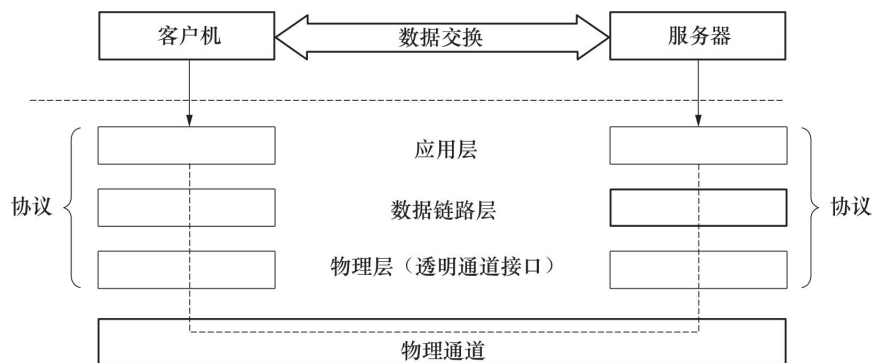


图 1 信息交换模型

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。

4.2 应用连接的数据交换

本部分中服务器和客户机之间是基于应用连接的数据交换，数据交换过程见图 2。

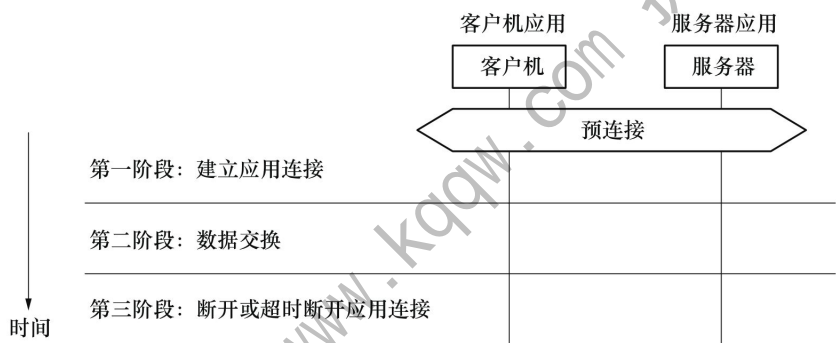


图 2 面向应用连接的数据交换过程

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。当客户机需要得到较高权限的服务器服务时，客户机必须发起建立较高权限的应用连接。

4.3 请求/响应类型的数据交换

本部分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

4.4 通知/确认类型的数据交换

本部分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

## 4.5 服务器模型

服务器模型见图 5。物理设备由若干逻辑设备构成，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（简称预连接对象）、若干个应用连接对象、若干个其他接口类对象。

逻辑设备 0 为必须具备的逻辑设备。

预连接对象为逻辑设备必须具备的对象。

应用连接对象为可选对象，根据访问权限所需进行增加。

其他接口类对象为可选对象，根据设备功能所需进行增加。

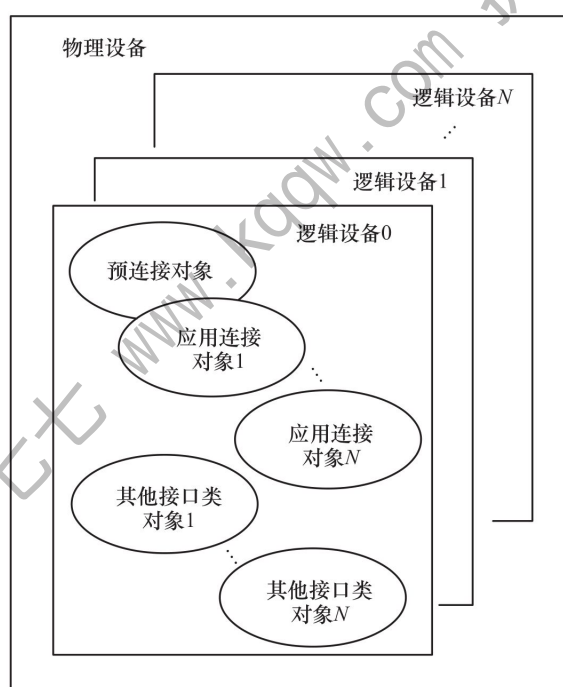


图5 服务器模型

## 5 数据链路层

### 5.1 帧结构

#### 5.1.1 帧格式

本部分采用异步式传输帧结构，定义见图 6。

起始字符（68H）	帧头
长度域L	
控制域C	
地址域A	
帧头校验HCS	
链路用户数据	APDU或APDU 分帧片段
帧校验FCS	帧尾
结束字符（16H）	

图 6 帧格式定义

5.1.2 长度域 L

长度域 L 由 2B 组成，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													

图 7 长度域定义

帧数据长度由 bit0～bit13 组成，采用 BIN 编码，是传输帧中除起始字符和结束字符之外的帧长度。

5.1.3 控制域 C

5.1.3.1 控制域格式定义

控制域 C 为 1B，按位或位的组合使用，定义见图 8。

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

图 8 控制域 C 定义

5.1.3.2 传输方向位及启动标识位

传输方向位：DIR=0 表示此帧是由客户机发出的；DIR=1 表示此帧是由服务器发出的。

启动标识位：PRM=0 表示此帧是由服务器发起的；PRM=1 表示此帧是由客户机发起的。

传输方向位（DIR）和启动标识位（PRM）组合意义见表 2。

表 2 传输方向位（DIR）和启动标识位（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

5.1.3.3 分帧标识位

分帧标识位为 0，表示此帧链路用户数据为完整 APDU；分帧标识位为 1，表示此帧链路用户数据为 APDU 片段，收齐所有片段按片段序号合并后为完整 APDU。

5.1.3.4 扰码标识位

扰码标识位为 0，表示此帧链路用户数据不加扰码；扰码标识位为 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。

5.1.3.5 功能码

功能码采用 BIN 编码，定义见表 3。

表 3 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	

5.1.4 地址域 A

5.1.4.1 地址域组成

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

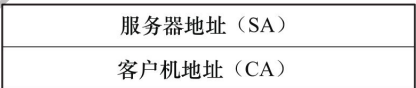


图 9 地址域定义

5.1.4.2 服务器地址（SA）

5.1.4.2.1 服务器地址定义

服务器地址由地址类型、逻辑地址、地址长度  $N$  及  $N$  字节地址组成，定义见图 10。



图 10 服务器地址定义

服务器地址第一个字节用 bit0~bit7 表示，该字节八位位组的最低位到最高位定义为：



- a) bit0~bit3: 地址的字节数, 取值范围为 0~15, 对应表示 1~16B 长度;
- b) bit4~bit5: 逻辑地址;
- c) bit6~bit7: 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。

#### 5.1.4.2.2 单地址

当服务器地址（SA）的地址类型为单地址时，其地址长度可变，用地址长度域表示，最长 16B，对应取值范围为 1…99999999999999999999999999999999，0 保留。其排列是高位在前，低位在后，十进制每两位占 1B，用 bit0～bit7 表示 8 位（位）组的最低位到最高位，各字节中 bit7～bit4 对应十进制高位，bit3～bit0 对应低位，为压缩 BCD 码方式，即每字节二进制高低各 4 位分别编码表示两个 0～9 的十进制数，且分别对应十进制数的高低两位。当排列的最后字节中的十进制低位（即 bit3～bit0）为 FH 时，表示为无效，即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。举例如下：

单地址=12345678 的排列见图 11。

LSB		LSB		LSB		LSB	
1	2	3	4	5	6	7	8
第1字节		第2字节		第3字节		第4字节	

图 11 单地址=12345678 的排列

单地址=123456789 的排列见图 12。

LSB		LSB		LSB		LSB		LSB	
1	2	3	4	5	6	7	8	9	F
第1字节		第2字节		第3字节		第4字节		第5字节	

图 12 单地址=123456789 的排列

#### 5.1.4.2.3 通配地址

当服务器地址的地址类型为通配地址时，其地址长度可变，由地址长度域表示。其排列是高位在前，低位在后，十进制每两位占 1B，用 bit0~bit7 表示八位（位）组的最低位到最高位，各字节中 bit7~bit4 对应十进制高位，bit3~bit0 对应低位，为压缩 BCD 码方式，即每字节二进制高低各 4 位分别编码表示两个 0~9 的十进制数或通配符 AH，且分别对应十进制数的高低两位。当排列的最后字节中的十进制低位（即 bit3~bit0）为 FH 时，表示为无效，即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。

通配符对应十进制按位使用，即：通配地址的某些十进制位为 AH，表示该位可为 0~9 的任意值，同时，凡不大于传输帧中通配地址所能表示的最大值的，且符合通配地址要求的服务器地址被选中。举例如下：

通配地址=12345678A，其排列见图 13，表示服务器地址不大于 99999999 的且符合 123456780~123456789 的服务器都需响应。

LSB		LSB		LSB		LSB		LSB	
1	2	3	4	5	6	7	8	A	F
第1字节		第2字节		第3字节		第4字节		第5字节	

图 13 通配地址=12345678A 的排列

#### 5.1.4.2.4 组地址

当服务器地址（SA）的地址类型为组地址时，同 5.1.4.2.2。组地址对系统中凡是属于该群组的服务器都有效，该服务器无须回答。

#### 5.1.4.2.5 广播地址

当服务器地址（SA）的地址类型为广播地址时，广播地址=AAH。广播地址对系统所有服务器都有效，所有服务器无须回答。

#### 5.1.4.3 客户机地址（CA）

客户机地址（CA）用 1B 无符号整数表示，取值范围 0~255，值为 0 表示不关注客户机地址。

#### 5.1.5 帧头校验（HCS）

帧头校验（HCS）为 2B，是对帧头部分除起始字符和 HCS 本身之外的所有字节的校验，校验算法见附录 A。

#### 5.1.6 链路用户数据

链路用户数据包含一个完整的应用层数据单元（APDU）字节序列或 APDU 的分帧片段，APDU 定义见 6.3.4。

#### 5.1.7 帧校验（FCS）

帧校验（FCS）为 2B，是对整帧除起始字符、结束字符和 FCS 本身之外的所有字节的校验，校验算法见附录 A。

### 5.2 字节格式

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

### 5.3 传输规则

#### 5.3.1 字节传输规则

若采用串行通信方式实现本地数据传输，则：

- a) 线路空闲状态为二进制 1。
- b) 发送数据时，在有效数据帧前加 4 个 FEH 作为前导码。
- c) 帧的字符之间无线路空闲间隔，两帧之间的线路空闲间隔最少需 33 位。
- d) 如按帧头校验 HCS 和帧校验 FCS 检出了差错，则两帧之间的线路空闲间隔最少需 33 位。

#### 5.3.2 分帧传输

##### 5.3.2.1 分帧传输规则

当一个 APDU 数据组帧长度超过协商的最大帧长度时，可采用分帧传输。采用分帧传输时，控制

域中分帧标识位置“1”。分帧数据接收端应对分帧传输进行逐条确认。

5.3.2.2 分帧传输格式定义

分帧传输时，链路层的链路用户数据为分帧传输帧，分帧传输帧格式定义见图 14。分帧传输的确认帧仅包含分帧格式域，不含 APDU 片段。

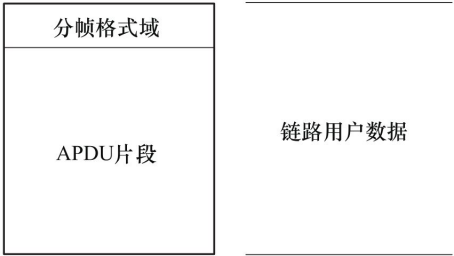


图 14 分帧传输帧格式定义

5.3.2.3 分帧格式域定义

分帧格式域为 2B，按位或位的组合使用，具体定义见图 15。

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

图 15 分帧格式域定义

bit0~bit11：表示分帧传输过程的帧序号，取值范围 0~4095，循环使用；

bit12~bit13：保留；

bit15=0，bit14=0：表示分帧传输数据起始帧；

bit15=1，bit14=0：表示分帧传输确认帧（确认帧不包含 APDU 片段域）；

bit15=0，bit14=1：表示分帧传输最后帧；

bit15=1，bit14=1：表示分帧传输中间帧。

5.3.2.4 分帧传输交换规则

分帧传输交互可由服务器或客户机任意一侧发起，也可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片段，采用分帧规则依次发送。

由服务器或客户机启动传输的数据分帧传输时序见图 16。

服务器响应客户机请求的数据分帧传输时序见图 17。

分帧的起始帧，block=0，type=0。

分帧传输确认帧，type=2，block 为最近一次收到正确的帧序号。对端在接收到该确认报文后应准备并传输下一个数据块，此数据块包含分帧序号 block=上次正确发送的帧序号+1。

分帧传输中间帧，type=3，block=收到确认帧的帧序号+1。

分帧的最后一帧，block=N（N 为实际包序号），type=1，分帧传输完毕。

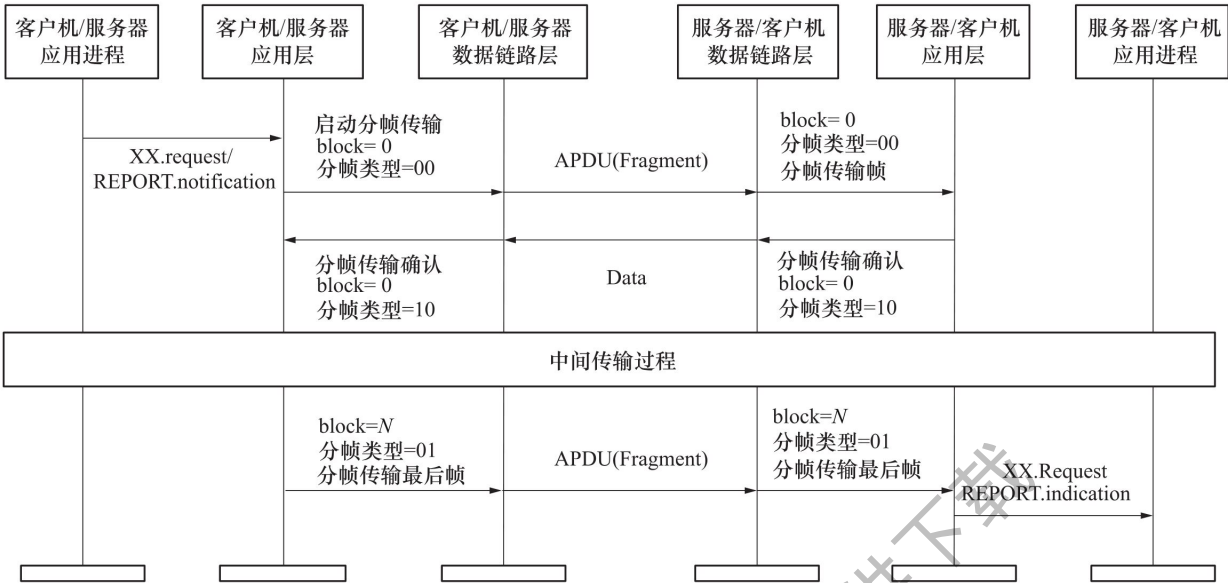


图 16 服务器/客户机启动传输的数据请求服务的分帧过程

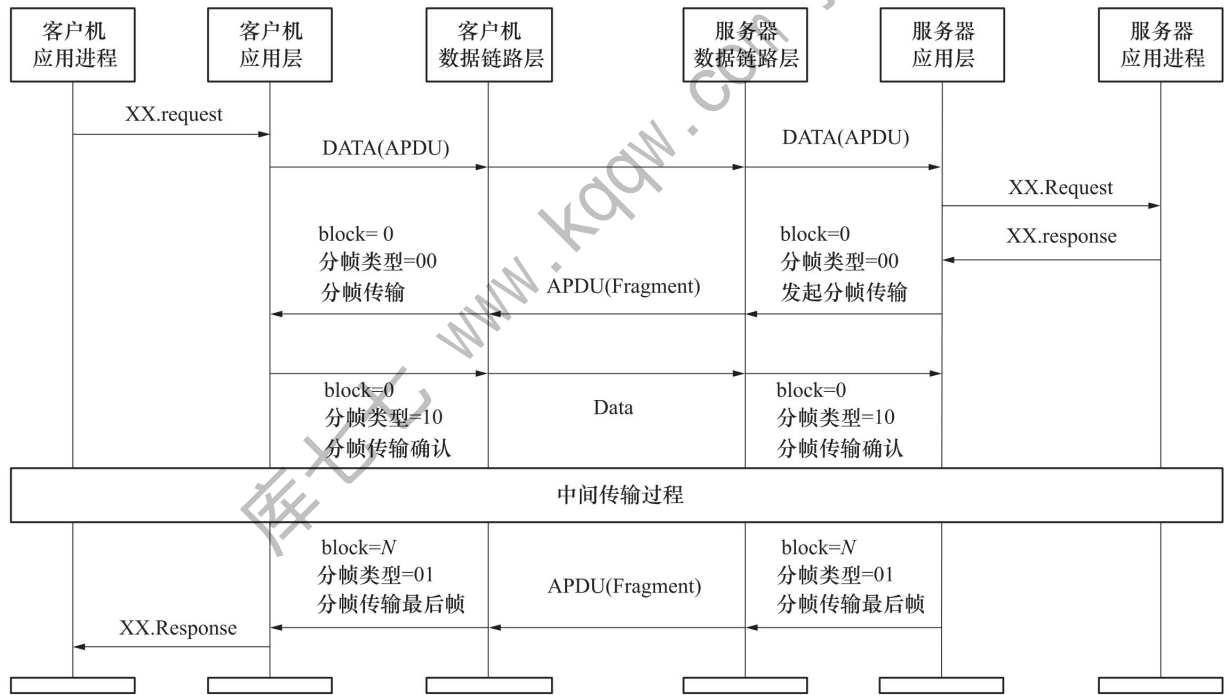


图 17 响应数据请求的服务分帧过程

6 应用层

6.1 应用层服务规范

6.1.1 应用层服务概述

6.1.1.1 应用层服务对象组件

应用层服务对象是构成客户机和服务器应用层的主要组件，它使用数据链路层提供的服务，服务

规范包含客户机和服务器应用进程在各自应用层的逻辑接口，并向应用进程提供服务。客户机和服务器的应用服务对象都包括预连接、应用连接和数据交换三个必备组件。

6.1.1.2 预连接

预连接服务适用于交换网络传输信道，如以太网、GPRS 等，当其完成物理连接，建立透明通道后，需要在此通道上建立预连接并进行管理。

预连接对应客户机和服务器各自应用层提供的服务见表 4。

表 4 应用层提供的预连接服务

服务名称	客户机应用层服务	服务器应用层服务
预连接	LINK (.indication, .response)	LINK (.request, confirm)

预连接服务（LINK）由服务器发起，客户机响应，LINK 服务包括：

- a) 登录：在完成物理连接，建立透明通道后，服务器应用进程按预连接配置参数向指定客户机发出登录请求，客户机应用进程给予确认，完成预连接。
- b) 心跳：服务器采用“心跳”的方式来保证预连接通道处于活动状态。
- c) 退出登录：在建立预连接后，不允许服务器主动断开。若要变更，需重新配置服务器预连接参数，服务器在重启后生效；或接收到客户机执行“复位”方法后，向原客户机发出退出登录指示，然后再按照新配参数执行新的预连接登录。

对于本地通信信道，如 RS-485、红外等，当物理连接建立时，默认预连接的通道即存在，不需要额外的建立以及预连接管理。

6.1.1.3 应用连接

6.1.1.3.1 应用连接窗口

为访问服务器的接口类对象，首先要建立一个应用连接，并创建一个可以相互通信的语境。这个语境主要包含应用语境的信息、使用身份验证机制的信息，以及其他需要的信息，这些信息包含在应用连接的接口类对象中。

服务器可以授予不同的访问权限给应用连接，访问权限涉及一组接口类对象，这组对象可以在给定的应用连接内被访问，即可视对象。

客户机可以通过读取应用连接对象的“可访问对象列表”属性而获得可视对象列表，即应用连接窗口，并利用应用连接对象所提供的方法，在已建立的应用连接内获得当前语境等更多的信息。

在预连接通道上，默认具有一个最低权限级别的应用连接窗口，即预建立的应用连接窗口，在此窗口内，客户机不需要进行应用连接协商及安全认证等便可访问该应用连接窗口的内容。

6.1.1.3.2 建立和断开应用连接

建立应用连接（CONNECT），由客户机向服务器发起，用于确认客户机和服务器双方通信的应用语境，包含协议一致性、功能一致性以及安全认证等内容。

服务器可同时支持若干个应用连接，互不干扰，但对同一个客户机，同时仅支持一个应用连接，当同一个客户机再次请求建立应用连接时，服务器如接受了客户机的再次请求，则前一个应用连接自动失效。

断开应用连接（RELEASE）用于正常断开一个已经建立的应用连接。由于不允许服务器提出正常

断开应用连接的请求，所以 RELEASE.request 服务只能由客户机提出，并且通常情况下，服务器不得拒绝此请求。

每一个应用连接在建立过程中，可以协商应用连接的静态超时时间，当连续无通信时间达到静态超时时间后，服务器将使用 RELEASE.notification 通知客户机，应用连接失效将被断开，此服务不需要客户机做任何响应。

应用连接对应客户机和服务器各自应用层提供的服务见表 5。

表 5 应用层提供的应用连接服务

服务名称	客户机应用层服务	服务器应用层服务
客户机建立应用连接	CONNECT (.request, .confirm)	CONNECT (.indication, .response)
客户机断开应用连接	RELEASE (.request, .confirm)	RELEASE (.indication, .response)
超时断开应用连接		RELEASE (.notification)

#### 6.1.1.3.3 预连接时建立的应用连接

预连接时建立的应用连接不需要使用 CONNECT 服务，即认为 CONNECT 已经完成，因此，预连接时建立的应用连接可以看成是在客户机和服务器之间完成预连接时应用连接已经存在，任何时候它都不能被断开，仅具有最低权限级别，窗口内容由服务器定义。这种应用连接简化了客户机和服务器之间数据交换，省掉了建立和断开应用连接阶段，仅有数据交换阶段。当客户机需要得到较高权限的服务器服务时，客户机必须发起建立较高权限的应用连接。

#### 6.1.1.4 数据交换

数据交换服务是用于客户机和服务器之间的数据交换，是通过逻辑名引用来访问接口对象的属性或方法。数据通信服务对应客户机和服务器各自应用层提供的服务见表 6。

表 6 应用层提供的数据交换服务

服务名称	客户机应用层服务	服务器应用层服务
读取	GET (.request, .confirm)	GET (.indication, .response)
设置	SET (.request, .confirm)	SET (.indication, .response)
操作	ACTION (.request, .confirm)	ACTION (.indication, .response)
上报	REPORT (.indication, .response)	REPORT (.notification, .confirm)
代理	PROXY (.request, .confirm)	PROXY (.indication, .response)

这些服务可分为两种通信类型：请求/响应类型，通知/确认类型。

请求/响应类数据交换服务是：读取（GET）、设置（SET）、操作（ACTION）、代理（PROXY）。

通知/确认类数据交换服务是：上报（REPORT）。

请求/响应类数据交换服务是通过客户机和服务器应用进程之间的数据交换来提供并完成的，即：客户机应用进程通过调用应用层的某个服务请求 XX.request，服务器应用层接收到客户机请求后向服务器应用进程发出服务指示 XX.indication，然后应用进程通过调用服务 XX.response 以响应客户机请求，客户机应用层接收到服务器响应后向客户机应用进程返回服务确认 XX.confirm。其正常服务顺序见图 18。



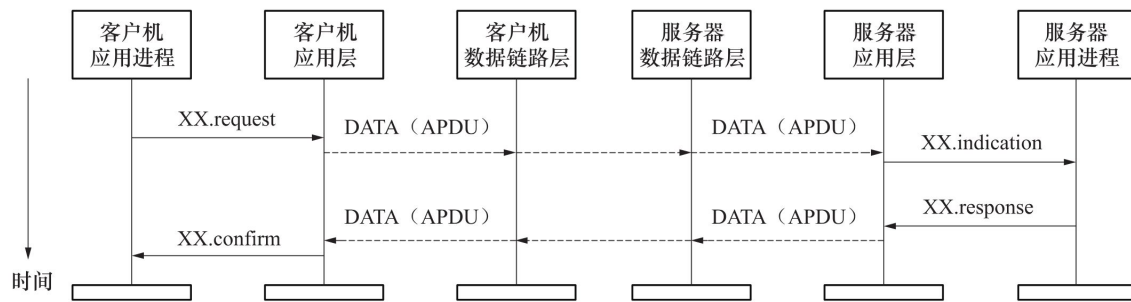


图 18 请求/响应类型的正常服务顺序

对于请求/响应类数据通信服务，在通信语境商定后，客户机和服务器的数据通信服务集是完全对等互补的，即：服务集相同，只是 XX.request 服务换成了 XX.indication 服务，XX.response 服务换成了 XX.confirm 服务。因此，一个 XX.request 的 APDU 与一个 XX.indication 的 APDU 对等，一个 XX.response 的 APDU 与一个 XX.confirm 的 APDU 对等。

通知/确认类数据交换服务也是通过客户机和服务器应用进程之间的数据交换来提供并完成的，即：在客户机向服务器定制了主动上报的情况下，服务器应用进程通过调用应用层服务 YY.notification，客户机应用层接收到服务器上报告后向客户机应用进程发出服务指示 YY.indication，然后客户机应用进程通过调用服务 YY.response 向服务器予以确认响应，服务器应用层接收到客户机确认后向服务器应用进程返回服务确认 YY.confirm。其正常服务顺序见图 19。

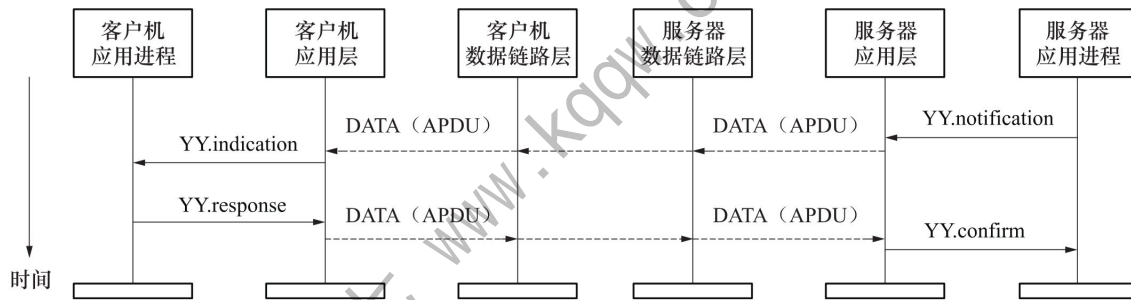


图 19 通知/确认类型的正常服务顺序

对于通知/确认类数据通信服务，在通信语境商定后，客户机和服务器的数据通信服务集也是完全对等互补的，即：服务集相同，只是 YY.notification 服务换成了 YY.indication 服务，YY.response 服务换成了 YY.confirm 服务。因此，一个 YY.notification 的 APDU 与一个 YY.indication 的 APDU 对等；一个 YY.response 的 APDU 与一个 YY.confirm 的 APDU 对等。

### 6.1.1.5 有关传输的时间标签

时间标签用于传输时序和时效性判断，其包括一个开始发送时间和一个允许传输延时时间。

允许传输延时时间，是指从开始发送至对方接收到能解析的完整的应用层数据单元之间所允许的传输延时时间。

对于请求/响应类数据交换服务，时间标签由客户机产生，随请求传送给服务器，服务器据此判决收到的请求的时序和时效性，如判别有效，响应收到的请求，并在响应中将接收到的时间标签返回客户机。

对于通知/确认类数据通信服务，时间标签由服务器产生，随通知传送给客户机，客户机据此判决收到的通知的时序和时效性，如判别有效，确认收到的通知，并在确认中将接收到的时间标签返回服务器。

时效性判断规则：在时间标签中允许传输延时时间大于零的前提下，如果接收方的当前时间与时间标

签中的开始发送时间之间的时差大于时间标签中的允许传输延时时间，则放弃处理；反之，则处理。

#### 6.1.1.6 有关服务器信息上报

##### 6.1.1.6.1 服务器上报服务

服务器上报服务（REPORT）是通过“注册-通知-撤销注册”的机制给客户机提供的一种系统级服务。客户机可通过 GET 服务查询出服务器支持的可注册后上报的服务集（如事件或定时数据上报等），并可根据系统需求通过 SET 服务以自定义形式注册部分或全部服务。注册成功后，服务器在检测到上报条件满足时（如产生了事件或定时上报时间到等），通过 REPORT.notification 服务及时通知客户机。

该服务默认对远程通道有效，本地通道提供同样服务应由服务器界面提供相关信息指示，并经配置后使用。

##### 6.1.1.6.2 服务器 APDU 的跟随上报信息域

服务器应用层数据单元（APDU）中的可选的跟随上报信息域，是当系统不适合或服务器不支持上报服务（REPORT）时，用于作为 ACD 标识事件上报方式的补充，以实现更及时上报客户机注册的上报信息。

该域同样默认对远程通道选择性有效，本地通道提供同样服务应由服务器界面提供相关信息指示，并经配置后使用。

#### 6.1.2 预连接

##### 6.1.2.1 LINK.request 服务

本服务由服务器应用进程调用，用于向远方客户机应用进程提出登录、心跳或退出登录三类预连接请求。

本服务用原语 LINK.request 表示，其服务原语及其参数说明见表 7。

表 7 LINK.request 服务原语及其参数说明

服务原语	参数说明
LINK.request ( 请求类型, 心跳周期, 请求时间 )	请求类型——登录、心跳、退出登录。 心跳周期——两次心跳请求的时间间隔。 请求时间——请求时服务器的时钟时间

##### 6.1.2.2 LINK.indication 服务

本服务由客户机应用层调用，用于向客户机应用进程指示接收到远方服务器提出的预连接请求。

本服务用原语 LINK.indication 表示，其服务原语及其参数说明同 LINK.request，见表 7。

##### 6.1.2.3 LINK.response 服务

本服务由客户机应用进程调用，用于向服务器应用进程响应预连接请求。

本服务用原语 LINK.response 表示，其服务原语及其参数说明见表 8。



表 8 LINK.response 服务原语及其参数说明

服务原语	参数说明
LINK.response ( 请求类型, 结果, 请求时间, 收到时间, 响应时间, 时间可信度 )	请求类型——登录、心跳、退出登录。 结果——用于表明请求是否成功或失败及其原因。 请求时间——LINK.request 的“请求时间”。 收到时间——接收到 LINK.indication 的时间。 响应时间——发出 LINK.response 的时间。 时间可信度——用于表明客户机时钟的准确性

6.1.2.4 LINK.confirm 服务

本服务由服务器应用层调用，用于向服务器应用进程指示接收到预连接请求的响应。  
本服务用原语 LINK.confirm 表示，其服务原语及其参数说明同 LINK.response，见表 8。

6.1.3 建立应用连接

6.1.3.1 CONNECT.request 服务

本服务由客户机应用进程调用，用于向远方服务器的应用进程提出建立应用连接请求。  
本服务用原语 CONNECT.request 表示，其服务原语及其参数说明见表 9。

表 9 CONNECT.request 服务原语及其参数说明

服务原语	参数说明
CONNECT.request ( 期望的应用层协议版本号, 期望的协议一致性块, 期望的功能一致性块, 客户机发送帧最大尺寸, 客户机接收帧最大尺寸, 客户机接收帧最大窗口尺寸, 客户机最大可处理 APDU 尺寸, 期望的应用连接超时时间, 认证机制信息 )	认证机制信息——包括以下内容： 1) 公共连接——不需要安全机制； 2) 一般密码——使用明文密码； 3) 对称加密——使用 ESAM 对称加密进行安全认证，并建立会话密钥； 4) 数字签名——使用 ESAM 非对称加密进行安全认证，并建立会话密钥

6.1.3.2 CONNECT.indication 服务

本服务由服务器应用层调用，用于向服务器应用进程指示接收到远方客户机提出的建立应用连接的请求。  
本服务用原语 CONNECT.indication 表示，其服务原语及其参数说明同 CONNECT.request，见表 9。

6.1.3.3 CONNECT.response 服务

本服务由服务器应用进程调用，用于向客户机应用进程返回请求结果。  
本服务用原语 CONNECT.response 表示，其服务原语及其参数说明见表 10。

表 10 CONNECT.response 服务原语及其参数说明

服务原语	参数说明
CONNECT.response ( 服务器厂商版本信息, 商定的应用层协议版本号, 商定的协议一致性块, 商定的功能一致性块, 服务器发送帧最大尺寸, 服务器接收帧最大尺寸, 服务器接收帧最大窗口尺寸, 服务器最大可处理 APDU 尺寸, 商定的应用连接超时时间, 认证响应 )	商定的应用连接超时时间——单位: s

## 6.1.3.4 CONNECT.confirm 服务

本服务由客户机应用层调用, 用于向客户机应用进程指示接收到建立应用连接请求的响应。

本服务用原语 CONNECT.confirm 表示, 其服务原语及其参数说明同 CONNECT.response, 见表 10。

## 6.1.4 断开应用连接

## 6.1.4.1 RELEASE.request 服务

本服务由客户机应用进程调用, 用于向远方服务器的应用进程提出断开应用连接请求。

本服务用原语 RELEASE.request 表示, 其服务原语及其参数说明见表 11。

表 11 RELEASE.request 服务原语及其参数说明

服务原语	参数说明
RELEASE.request ( )	无参数

## 6.1.4.2 RELEASE.indication 服务

本服务由服务器应用层调用, 用于向服务器应用进程指示接收到远方客户机提出的断开应用连接的请求。

本服务用原语 RELEASE.indication 表示, 其服务原语及其参数说明同 RELEASE.request, 见表 11。

## 6.1.4.3 RELEASE.response 服务

本服务由服务器应用进程调用, 用于向客户机应用进程返回请求结果。

本服务用原语 RELEASE.response 表示, 其服务原语及其参数说明见表 12。

表 12 RELEASE.response 服务原语及其参数说明

服务原语	参 数 说 明
RELEASE.response ( 结果 )	结果——请求的结果, 通常情况下, 服务器不得拒绝此请求

6.1.4.4 RELEASE.confirm 服务

本服务由客户机应用层调用，用于向客户机应用进程指示接收到断开应用连接请求的响应。  
本服务用原语 RELEASE.confirm 表示，其服务原语及其参数说明同 RELEASE.response，见表 12。

6.1.4.5 RELEASE.notification 服务

本服务由服务器应用进程调用，用于通知客户机应用进程连接因超时而失效将被断开。本服务不需要客户机做任何响应。

本服务用原语 RELEASE.notification 表示，其服务原语及其参数说明见表 13。

表 13 RELEASE.notification 服务原语及其参数说明

服务原语	参 数 说 明
RELEASE.notification ( 应用连接建立时间， 服务器当前时间 )	应用连接建立时间——建立应用链接时的终端时间； 服务器当前时间——发出连接断开的终端时间

6.1.5 读取

6.1.5.1 GET.request 服务

本服务由客户机应用进程调用，用于向远方服务器的应用进程请求服务器的若干个接口类对象的所有属性值。

本服务用原语 GET.request 表示，其服务原语及其参数说明见表 14。

表 14 GET.request 服务原语及其参数说明

服务原语	参 数 说 明
GET.request ( 请求类型， 对象属性描述符， {对象属性描述符，} 数据块序号 )	请求类型——用于区分不同的读取请求，分为以下几种类型： 1) 读取一个对象属性； 2) 读取若干个对象属性； 3) 读取一个记录型对象属性； 4) 读取若干个记录型对象属性； 5) 读取分帧响应的下一个数据块，仅在被请求的数据响应不能在一个 GET.response APDU 中传输时才使用。 对象属性描述符——仅当“请求类型”为 1) ~4) 时才出现，用于表明要读取的对象属性，分为一般型、记录型两种。 数据块序号——用于表明正确接收到的最近一次数据块的序号，仅当“请求类型”为 5) 时出现

6.1.5.2 GET.indication 服务

本服务由服务器应用层调用，用于向服务器应用进程指示接收到远方客户机 GET.request 服务请求。本服务用原语 GET.indication 表示，其服务原语及其参数同 GET.request，见表 14。

6.1.5.3 GET.response 服务

本服务由服务器应用进程调用，用于对应 GET.indication 向客户机应用进程返回请求结果。  
本服务用原语 GET.response 表示，其服务原语及其参数说明见表 15。

表 15 GET.response 服务原语及其参数说明

服务原语	参 数 说 明
GET.response ( 响应类型, 读取结果, {读取结果,} 数据块序号 )	<p>响应类型——用于表明响应是否包含对应 GET.request 服务调用的响应, 或只包含部分响应, 对应请求类型, 响应类型分为以下几种类型:</p> <ol style="list-style-type: none"> <li>1) 读取一个对象属性的响应;</li> <li>2) 读取若干个对象属性的响应;</li> <li>3) 读取一个记录型对象属性的响应;</li> <li>4) 读取若干个记录型对象属性的响应;</li> <li>5) 分帧响应一个数据块, 分帧方式按请求对象属性分成能自解释的若干独立响应, 且这一类根据 GET.request 请求类型还分为一般型、记录型两种对象属性。</li> </ol> <p>读取结果——用于表明读取请求的响应结果, 包括对象属性描述符及其数值, 对象属性为一般型、记录型两种。如“读取结果”的编码形式不适合在一个 APDU 中传输, 则它应采用分帧响应, 即“响应类型”为 5)。</p> <p>6) 数据块序号——用于表明本 APDU 中数据块的序号, 仅在“响应类型”为 5) 时出现</p>

#### 6.1.5.4 GET.confirm 服务

本服务由客户机应用层调用, 用于向客户机应用进程指示接收到服务器 GET.response APDU。

本服务用原语 GET.confirm 表示, 其服务原语及其参数说明同 GET.response。

#### 6.1.6 设置

##### 6.1.6.1 SET.request 服务

本服务由客户机应用进程调用, 用于向远方服务器的应用进程设置服务器的若干个接口类对象的一个或所有的属性值。

本服务用原语 SET.request 表示, 其服务原语及其参数说明见表 16。

表 16 SET.request 服务原语及其参数说明

服务原语	参 数 说 明
SET.request ( 请求类型, 对象属性描述符及其数值, {对象属性描述符及其数值,} 对象属性描述符, {对象属性描述符,} )	<p>请求类型——用于区分不同的设置请求, 分为以下几种类型:</p> <ol style="list-style-type: none"> <li>1) 设置一个对象属性请求;</li> <li>2) 设置若干个对象属性请求;</li> <li>3) 设置后读取若干个对象属性请求。</li> </ol> <p>对象属性描述符及其数值——用于表明要设置的对象属性及其设置数值。</p> <p>对象属性描述符——仅当“请求类型”为 3) 时才出现, 用于表明在设置后要读取的对象属性, 对象属性仅为一般型</p>

##### 6.1.6.2 SET.indication 服务

本服务由服务器应用层调用, 用于向服务器应用进程指示接收到远方客户机 SET.request 服务请求。本服务用原语 SET.indication 表示, 其服务原语及其参数说明同 SET.request。

##### 6.1.6.3 SET.response 服务

本服务由服务器应用进程调用, 用于对应 SET.indication 向客户机应用进程返回请求结果。

本服务用原语 SET.response 表示, 其服务原语及其参数说明见表 17。

表 17 SET.response 服务原语及其参数说明

服务原语	参 数 说 明
SET.response ( 响应类型, 设置结果, {设置结果,} 读取结果, {读取结果,} )	响应类型——用于表明响应所对应的 SET.request 服务的请求类型, 对应请求类型, 响应类型分为以下几种类型: 1) 设置一个对象属性的确认信息; 2) 设置若干个对象属性的确认信息; 3) 设置若干个对象属性的确认信息以及读取若干个对象属性的响应。 设置结果——用于表明设置的执行结果, 包括设置的对象属性描述符及其结果。 读取结果——仅在“响应类型”为 3) 时出现, 用于表明设置后读取属性的响应结果, 包括设置后读取的对象属性描述符及其数据, 对象属性仅为一般型

6.1.6.4 SET.confirm 服务

本服务由客户机应用层调用, 用于向客户机应用进程指示接收到服务器 SET.response APDU。

本服务用原语 SET.confirm 表示, 其服务原语及其参数说明同 SET.response。

6.1.7 操作

6.1.7.1 ACTION.request 服务

本服务由客户机应用进程调用, 用于调用远方服务器应用进程中的若干个接口类对象的若干个方法。

本服务用原语 ACTION.request 表示, 其服务原语及其参数说明见表 18。

表 18 ACTION.request 服务原语及其参数说明

服务原语	参 数 说 明
ACTION.request ( 请求类型, 对象方法描述符及参数, {对象方法描述符及参数,} 对象属性描述符, {对象属性描述符,} )	请求类型——用于区分不同的操作请求, 分为以下几种类型: 1) 操作一个对象方法请求; 2) 操作若干个对象方法请求; 3) 操作若干个对象方法后读取若干个对象属性请求。 对象方法描述符及参数——用于表明要操作的对象方法及操作该方法的参数。 对象属性描述符——仅当“请求类型”为 3) 时才出现, 用于表明操作执行后要读取的对象属性, 对象属性仅为一般型

6.1.7.2 ACTION.indication 服务

本服务由服务器应用层调用, 用于向服务器应用进程指示接收到远方客户机 ACTION.request 服务请求。

本服务用原语 ACTION.indication 表示, 其服务原语及其参数说明同 ACTION.request。

6.1.7.3 ACTION.response 服务

本服务由服务器应用进程调用, 用于对应 ACTION.indication 向客户机应用进程返回请求结果。

本服务用原语 ACTION.response 表示, 其服务原语及其参数说明见表 19。

表 19 ACTION.response 服务原语及其参数说明

服务原语	参 数 说 明
ACTION.response ( 响应类型, 操作结果, {操作结果,} 读取结果, {读取结果,} )	<p>响应类型——用于表明响应所对应的 ACTION.request 服务的请求类型, 对应请求类型, 响应类型分为以下几种类型:</p> <p>1) 操作一个对象方法的响应;</p> <p>2) 操作若干个对象方法的响应;</p> <p>3) 操作若干个对象方法后读取若干个对象属性的响应。</p> <p>操作结果——用于表明调用对象方法的执行结果, 包括对象方法描述符及其结果。</p> <p>读取结果——仅在“响应类型”为 3) 时出现, 用于表明操作执行后要读取的对象属性的响应结果, 包括对象属性描述符及其数据, 对象属性仅为一般型</p>

## 6.1.7.4 ACTION.confirm 服务

本服务由客户机应用层调用, 用于向客户机应用进程指示接收到服务器 ACTION.response APDU。

本服务用原语 ACTION.confirm 表示, 其服务原语及其参数说明同 ACTION.response。

## 6.1.8 上报

## 6.1.8.1 REPORT.notification 服务

本服务由服务器应用进程调用, 用于向远方客户机应用进程上报信息, 该信息是由客户机通过注册方式预定的。

本服务用原语 REPORT.notification 表示, 其服务原语及其参数说明见表 20。

表 20 REPORT.notification 服务原语及其参数说明

服务原语	参 数 说 明
REPORT.notification ( 通知类型, 对象属性描述符及其数值, {对象属性描述符及其数值,} )	<p>通知类型——用于区分不同的上报通知, 分为以下几种类型:</p> <p>1) 上报若干个对象属性;</p> <p>2) 上报若干个记录型对象属性。</p> <p>对象属性描述符及其数值——用于表明上报的信息</p>

## 6.1.8.2 REPORT.indication 服务

本服务由客户机应用层调用, 用于向客户机应用进程指示接收到远方服务器 REPORT.notification 服务通知。

本服务用原语 REPORT.indication 表示, 其服务原语及其参数说明同 REPORT.notification。

## 6.1.8.3 REPORT.response 服务

本服务由客户机应用进程调用, 用于对应 REPORT.indication 向服务器应用进程返回确认结果。

本服务用原语 REPORT.response 表示, 其服务原语及其参数说明见表 21。

表 21 REPORT.response 服务原语及其参数说明

服务原语	参 数 说 明
REPORT.response ( 响应类型, 确认结果, {确认结果,} )	<p>响应类型——用于表明响应所对应的 REPORT.indication 服务的通知类型, 对应通知类型, 响应类型分为以下几种类型:</p> <p>1) 上报若干个对象属性的响应;</p> <p>2) 上报若干个记录型对象属性的响应。</p> <p>确认结果——用于表明上报的确认结果, 为被确认的对象属性描述符, 对象属性为一般型和记录型两种</p>

6.1.8.4 REPORT.confirm 服务

本服务由服务器应用层调用，用于向服务器应用进程指示接收到客户机 REPORT.response APDU。  
本服务用原语 REPORT.confirm 表示，其服务原语及其参数说明同 REPORT.response。

6.1.9 代理

6.1.9.1 PROXY.request 服务

本服务由客户机应用进程调用，用于向远方服务器（代理服务器）的应用进程提出代理请求。  
本服务用原语 PROXY.request 表示，其服务原语及其参数说明见表 22。

表 22 PROXY.request 服务原语及其参数说明

服务原语	参 数 说 明
PROXY.request ( 请求类型， 目标服务器地址/端口， {目标服务器地址，} 对象属性描述符， {对象属性描述符，} 对象属性描述符及其数值， {对象属性描述符及其数值，} 对象方法描述符， {对象方法描述符，} 代理相关参数， {代理相关参数，} 透明命令， )	<p>请求类型——用于区分不同的代理请求，分为以下几种类型：</p> <ol style="list-style-type: none"><li>1) 代理读取若干个服务器的若干个对象属性；</li><li>2) 代理读取一个服务器的一个记录型对象属性；</li><li>3) 代理设置若干个服务器的若干个对象属性；</li><li>4) 代理设置后读取若干个服务器的若干个对象属性；</li><li>5) 代理操作若干个服务器的若干个对象方法；</li><li>6) 代理操作后读取若干个服务器的若干个对象方法和属性；</li><li>7) 代理透明转发命令。</li></ol> <p>目标服务器地址/端口——用于表明代理的目标服务器地址，如 PROXY.request 的目标服务器地址采用了通配地址，响应时要分解为确定的服务器单地址，即 PROXY.response 的目标服务器地址皆为单地址。</p> <p>对象属性描述符——用于表明要代理读取的对象属性，当“请求类型”为 1)、4)、6) 时，对象属性仅为一般型，当“请求类型”为 2) 时，对象属性为记录型。</p> <p>对象属性描述符及其数值——用于表明要代理设置的对象属性，仅当“请求类型”为 3)、4) 时才出现，对象属性仅为一般型。</p> <p>对象方法描述符——用于表明要代理操作的对象方法，仅当“请求类型”为 5)、6) 时才出现。</p> <p>代理相关参数——与代理请求相关的所需参数。</p> <p>透明命令——用于表明带来透明转发的命令，仅当请求类型为代理透明转发命令时才出现</p>

6.1.9.2 PROXY.indication 服务

本服务由服务器应用层调用，用于向服务器应用进程指示接收到远方客户机 PROXY.request 服务请求。

本服务用原语 PROXY.indication 表示，其服务原语及其参数同 PROXY.request。

6.1.9.3 PROXY.response 服务

本服务由服务器应用进程调用，用于对应 PROXY.indication 向客户机应用进程返回代理请求的代理执行结果。

本服务用原语 PROXY.response 表示，其服务原语及其参数说明见表 23。

#### 6.1.9.4 PROXY.confirm 服务

本服务由客户机应用层调用，用于向客户机应用进程指示接收到服务器 PROXY.response APDU。  
本服务用原语 PROXY.confirm 表示，其服务原语及其参数同 PROXY.response。

表 23 PROXY.response 服务原语及其参数说明

服务原语	参 数 说 明
PROXY.response ( 响应类型, 目标服务器地址/端口, {目标服务器地址,} 读取结果, {读取结果,} 设置结果, {设置结果,} 操作结果, {操作结果,} 透明转发结果, {透明转发结果,} )	<p>响应类型——用于表明响应所对应的 PROXY.request 服务的请求类型，对应请求类型，响应类型分为以下几种类型：</p> <ol style="list-style-type: none"> <li>1) 代理读取若干个服务器的若干个对象属性的响应；</li> <li>2) 代理读取一个服务器的一个记录型对象属性的响应；</li> <li>3) 代理设置若干个服务器的若干个对象属性的确认；</li> <li>4) 代理设置后读取若干个服务器的若干个对象属性的确认和响应；</li> <li>5) 代理操作若干个服务器的若干个对象方法的确认；</li> <li>6) 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确认和响应；</li> <li>7) 代理透明转发命令的响应。</li> </ol> <p>目标服务器地址/端口——用于表明代理的目标服务器地址。</p> <p>读取结果——当“响应类型”为 1)、2)、4)、6) 时才出现，用于表明代理读取请求的响应结果，包括对象属性描述符及其数值，对象属性对应请求类型为一般型或记录型两种。</p> <p>设置结果——仅当“响应类型”为 3)、4) 时才出现，用于表明代理设置的执行结果，包括设置的对象属性描述符及其结果，对象属性仅为一般型。</p> <p>操作结果——仅在“响应类型”为 5)、6) 时出现，用于表明代理操作对象方法的执行结果，包括对象方法描述符及其操作结果。</p> <p>透明转发结果——仅在响应类型为代理透明转发命令的响应时出现，用于表明代理透明转发命令的执行结果，包括透明转发的应答数据</p>

## 6.2 应用层协议规范

### 6.2.1 建立/断开应用连接

#### 6.2.1.1 建立应用连接

建立应用连接借助于 CONNECT.request /.indication /.response /.confirm 服务。

客户机应用进程应首先调用 CONNECT.request 请求服务，调用该服务前预连接已经建立。客户机应构造一个 CONNECT-Request APDU。该 CONNECT-Request APDU 是发送给服务器应用层的第一个报文。

服务器应用层从接收到的 CONNECT-Request APDU 中提取出来的适当参数调用 CONNECT.indication 服务原语向服务器应用进程发出指示。

服务器应用进程分析接收到的 CONNECT.indication 原语，并且决定是否接受提出的应用连接请求，通过核实之后，服务器应用进程应调用 CONNECT.response 服务原语，表明接受或不接受提出的连接请求。如果成功，服务器应构造相应的 CONNECT-Response APDU，并通过现有的预连接通道发送给远方客户机应用层。从这一时刻起，服务器能够在该连接内接收数据通信服务请求，发送相应的响应。至此，应用连接建立完毕，服务器进入数据通信阶段。

如果服务器不能接受连接请求，服务器应用层应构造 CONNECT-Response APDU，其中包含拒绝连接的状态以及原因，发送至远方客户机应用层。

在客户机侧，提取接收到的 CONNECT-Response APDU 中的参数，并通过 CONNECT.confirm 服务原语发送给客户机应用进程，如连接请求被接受，从这时刻起，在协商应用的语境中，应用连接建立完成。



建立应用连接的时序图见图 20。

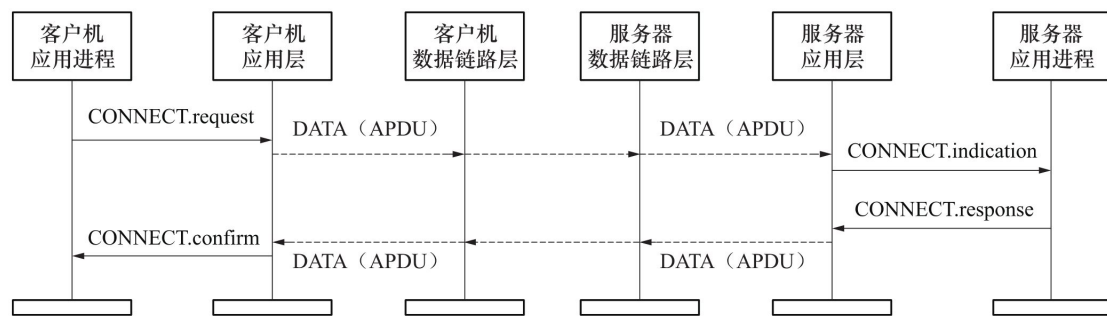


图 20 建立应用连接的时序图

6.2.1.2 断开应用连接

6.2.1.2.1 概述

现有应用连接能够被正常断开或超时被断开，正常断开由客户机应用进程启动，通知服务器侧，请求断开当前应用连接。

超时被断开意味着连接被异常终止，当应用层连续无通信时长超过语境约定的超时时间，应用连接将被断开，此服务只能由服务器应用进程启动。

6.2.1.2.2 正常断开应用连接

正常断开应用连接总是由客户机应用通过调用 RELEASE.request 的服务启动。根据协议生成一个 RELEASE-Request APDU，通过低层支持协议发送到服务器侧。

服务器应用层把接收到的 RELEASE-Request 解释为应用连接的断开请求，并且通过 RELEASE.indication 服务原语向服务器应用进程指示该请求。

服务器应用进程应接受断开请求并调用 RELEASE.response 服务（服务器不能拒绝客户机的断开连接请求）。

请求断开应用连接的时序图见图 21。

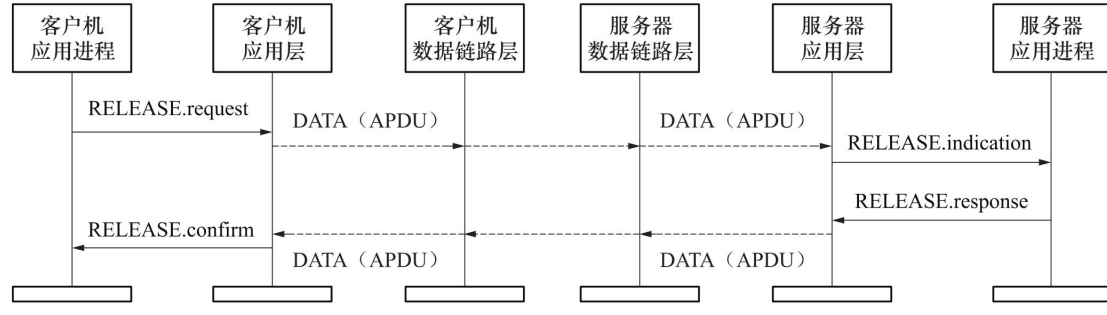


图 21 请求断开应用连接的时序图

6.2.1.2.3 通信超时导致应用连接被断开

当应用连接建立后（不包括预建立时建立的应用连接），如果连续无数据传输服务时长（不包括预连接管理服务）达到会话语境约定的超时时间后（造成通信超时的原因涵盖包括物理层故障在内的任何原因），服务器应用进程将调用 RELEASE.notification 服务，通知客户机此连接将被断开，客户机不需要做任何响应。

超时断开应用连接的时序图见图 22。

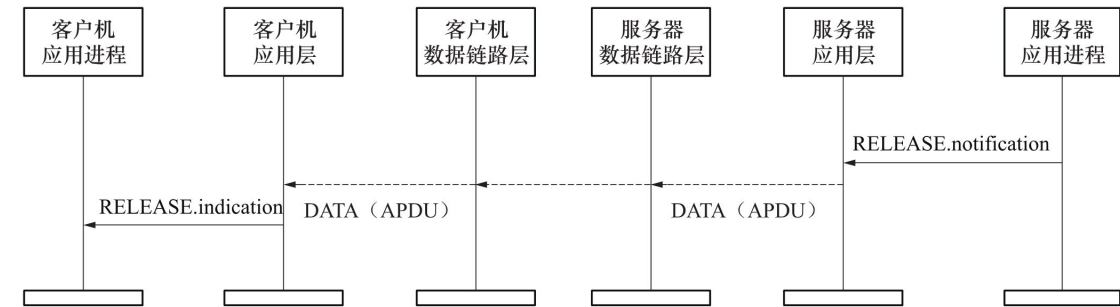


图 22 超时断开应用连接的时序图

6.2.2 请求/响应数据交换

6.2.2.1 短帧的数据交换

短帧，特指完整的一帧 APDU 长度不超过会话语境约定的发送或接收数据长度；长帧，特指一帧 APDU 长度超过会话语境约定的发送或接收数据长度，需要采取分帧传输。请求/响应数据交换服务包括：

- a) 读取 (.request/.indication/.response/.confirm)；
- b) 设置 (.request/.indication/.response/.confirm)；
- c) 操作 (.request/.indication/.response/.confirm)；
- d) 代理 (.request/.indication/.response/.confirm)。

读取和设置服务用于引用接口对象实例的属性，操作服务用于引用接口对象的方法，代理服务用于引用远方服务器的对象属性或方法。上述服务，在采用短帧数据交换时，具有相同的时序，在此一并描述。

服务器的应用进程一旦接收到数据通信服务指示，应从合法性、可行性等方面检查能否提供该服务，如果一切都正确，服务器应用进程应在本地使用相应的具体对象提供所请求的服务。服务器应用层应生成一个适当的.response 报文，包含.request 的执行结果，发送到客户机侧，一个交互流程完成。

短帧的数据交换时序图见图 23。

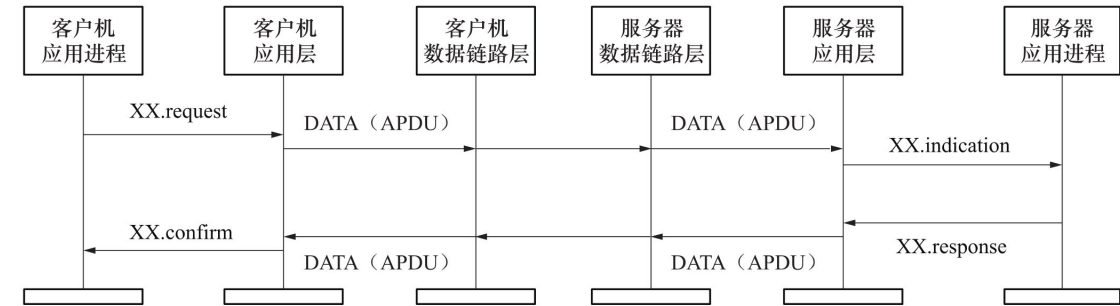


图 23 短帧的数据交换时序图

6.2.2.2 长帧的数据交换

6.2.2.2.1 长帧传输的读取服务

读取服务的分帧传输规范仅适用于 GET.response 服务原语中的数据。

在 GET.response 服务中多重属性引用时，其服务参数编码形式的长度不能超过发送帧最大尺寸。

如果 GET.response 服务参数编码长度超过最大尺寸，应使用分帧服务进行传输。

服务器一旦接收到一个 GET.request，服务器应用进程就组装所请求的数据，如果这些数据能够放在一个 APDU 中，服务器应用进程应调用对应短帧类型的 Get.response 服务，其结果参数包含所请求属性的值。

如果数据长度超过发送帧最大尺寸，则应使用分帧传输。分帧传输有两种可选模式，一种为单帧可自解析模式，另一种为不可自解析模式。不可自解析传输模式必须在所有数据片段接收完毕后，才能正确解析相应的数据。单帧可自解析的分帧方式，每一帧中必须包含完整的属性数据单元。

仅 GET.response 服务支持单帧可自解析模式，单帧自解析分帧传输时序见图 24。数据块传输模式见 5.3.2。

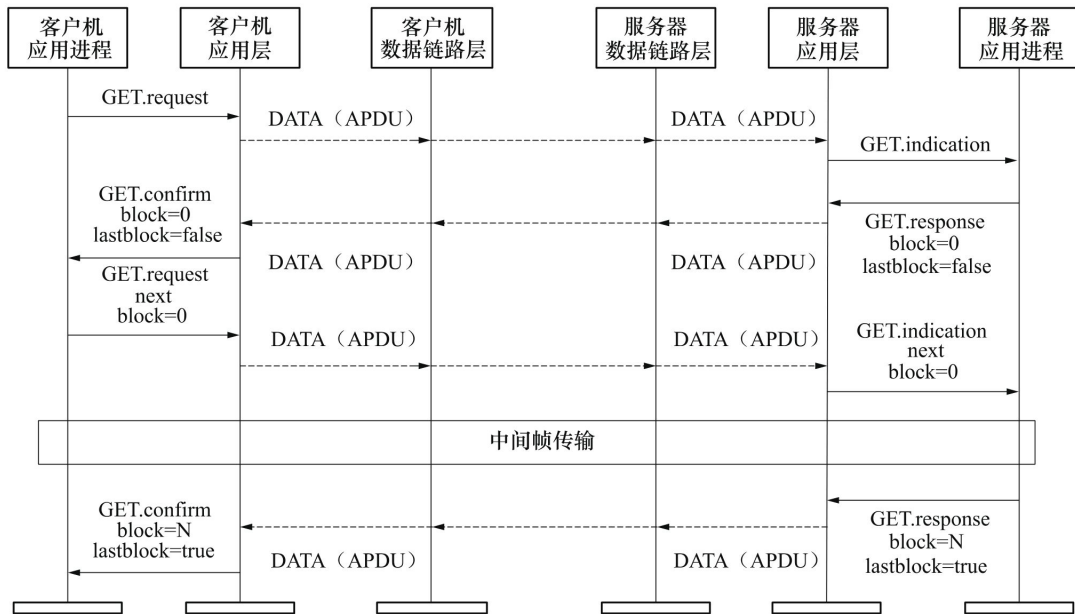


图 24 单帧自解析分帧传输时序图

分帧响应 GetResponseNext 的 APDU 数据域，有分帧序号 (block)、末帧标记 (lastblock) 两个关键的数据字段。分帧的第一帧数据：block (long-unsigned) = 0, lastblock (BOOLEAN) = False；分帧的最后一帧数据，block=N (N 为实际包序号)，lastblock=True，分帧传输完毕。

客户机一旦接收到 GetResponseNext，客户机应用进程知道所请求的响应数据已经超过接收帧最大尺寸，并准备处理后续帧传输，它应保存所接收的 APDU 的数据内容，并调用 GetRequestNext 服务对所接收到的数据块进行确认。

服务器在接收到该确认报文后应准备并传输下一个数据块，此数据块包含分帧序号 block=1，这样的数据块交换和确认正常继续下去，直到发送完最后一个数据块，此时 response 的 lastblock (BOOLEAN) 参数应置为 “True”，并且客户机不对该数据块进行确认，在接收到最后一个数据块后，GET 服务的分帧传输过程完成。

如果分帧传输期间发生以下差错，传输过程将终止：

- 服务器出于任何原因不能提供下一个数据块。这种情况下，服务器应发送一个 GetResponseNext 的 APDU，将 lastblock 参数置为 “True”，block 置为客户机所期望的值 (接收到的 block+1)，数据域 Result 参数包含一个 DAR 参数，用来指明差错原因。
- 服务器接收到 GetRequestNext 类型的 GET.indication，block 参数的值与服务器上一次发送的 block 参数值不相等。服务器对这种情况的解释为客户机期望终止正在进行的传输，服务器不再发送下一个数据块给客户机，而是返送一个 GetResponseNext APDU，将 lastblock 参数置为

“True”，block 为接收到的 block 参数值，数据域参数为 DAR=分帧传输已取消。

——服务器在没有进行分帧传输时接收到 GetRequestNext 类型的 GET.indication。在这种情况下，使用 GetResponseNext APDU 进行响应。将 lastblock 参数置为“True”，block 为接收到的 block 参数值，数据域参数为 DAR=不处于分帧传输状态。

在分帧传输期间，所有 APDU 中的 Invoke-Id 和 Priority 参数的值相同。如果在分帧传输期间，接收到其他服务请求，则按照优先级原则进行服务。

#### 6.2.2.2 长帧传输的其他服务

仅 GET.response 服务支持单帧可自解析模式，其他类型服务的数据域长度超过发送帧最大尺寸时，应使用分帧服务进行分帧传输。分帧服务传输时序参见 5.3.2。

#### 6.2.3 上报/确认数据交换

上报/确认类服务用于服务器主动发起、传送事件或者其他定时任务数据到客户机。在 REPORT.notification 服务中，其服务参数编码形式的长度不能超过发送帧最大尺寸。如果数据长度超出，应使用分帧服务进行传送。

客户在收到 REPORT.indication 指示时，应使用 REPORT.response 进行确认，在服务器收到 REPORT.confirm 时，方可认为主动上报发送成功。如果在约定的超时时间内未收到确认，将再次发起 REPORT.notification，APDU 中的 Invoke-Id 和 Priority 参数的值保持不变，在达到约定的最多重复次数后，如仍未收到确认帧，则放弃该 APDU 的主动上报。

上报/确认类服务，可以在服务器侧任意通信端口发起（包括本地端口及远程端口，根据配置决定），确认状态与上报的通信端口相关。只有在端口预连接正确建立的前提下，服务器方可发起主动上报。

REPORT 服务时序见图 25。

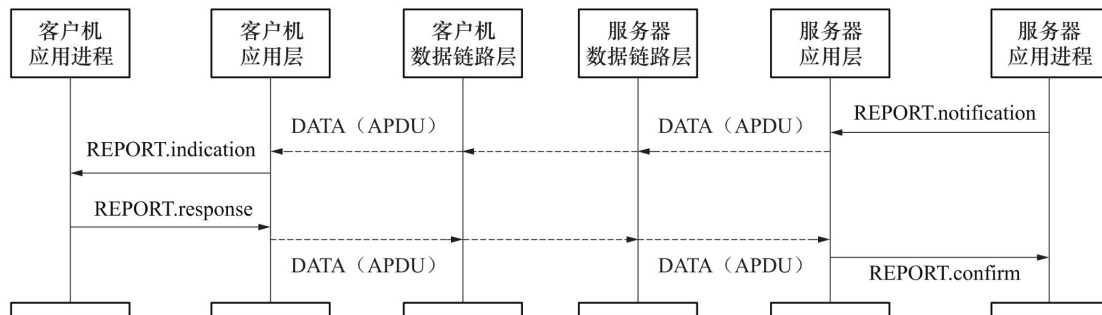


图 25 REPORT 服务时序图

### 6.3 应用层数据单元规范

#### 6.3.1 概述

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。

#### 6.3.2 通用数据类型

通用数据类型见表 24。

表 24 通用数据类型

类型描述	标记	定义	数值范围
NULL	0	空	
array	1	SEQUENCE OF Data (见 6.3.3.1) 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 6.3.3.1) 结构的元素在对象属性或方法的描述中定义	
bool	3	布尔值 (BOOLEAN)	1 或 0
bit-string	4	比特位串 (BIT STRING)	
double-long	5	32bit 整数 (Integer32)	$-2^{31} \sim 2^{31} - 1$
double-long-unsigned	6	32bit 正整数 (double-long-unsigned)	$0 \sim 2^{32} - 1$
保留	7、8		
octet-string	9	8 位 (位) 组 (字节) 串 (OCTET STRING)	
visible-string	10	ASCII 字符串 (VisibleString)	
保留	11		
UTF8-string	12	UTF-8 编码的字符串	
保留	13、14		
integer	15	8bit 整数 (integer)	$-128 \sim 127$
long	16	16bit 整数 (long)	$-32768 \sim 32767$
unsigned	17	8bit 正整数 (Unsigned8)	$0 \sim 255$
long-unsigned	18	16bit 正整数 (Unsigned16)	$0 \sim 65535$
保留	19		
long64	20	64bit 整数 (Integer64)	$-2^{63} \sim 2^{63} - 1$
long64-unsigned	21	64bit 正整数 (Unsigned64)	$0 \sim 2^{64} - 1$
enum	22	枚举的元素在对象属性或方法的描述中定义	$0 \sim 255$
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 6.3.3.24	
OAD	81	见 6.3.3.5	
ROAD	82	见 6.3.3.6	
OMD	83	见 6.3.3.12	
TI	84	见 6.3.3.22	

表 24（续）

类型描述	标记	定义	数值范围
TSA	85	见 6.3.3.13	
MAC	86	见 6.3.3.15	
RN	87	见 6.3.3.18	
Region	88	见 6.3.3.23	
Scaler_Unit	89	见 6.3.3.14	
RSD	90	见 6.3.3.7	
CSD	91	见 6.3.3.9	
MS	92	见 6.3.3.10	
SID	93	见 6.3.3.16	
SID_MAC	94	见 6.3.3.17	
COMDCB	95	见 6.3.3.29	
RCSD	96	见 6.3.3.8	
保留	97~255		

6.3.3 数据类型定义

6.3.3.1 Data 数据类型

Data 数据类型定义及说明见表 25。

表 25 Data 数据类型定义及说明

数据类型定义	说明
<div><div>Data ::= CHOICE</div><div>{</div><div><div>NULL</div><div>[0] ,</div></div><div><div>array</div><div>[1] ,</div></div><div><div>structure</div><div>[2] ,</div></div><div><div>bool</div><div>[3] ,</div></div><div><div>bit-string</div><div>[4] ,</div></div><div><div>double-long</div><div>[5] ,</div></div><div><div>double-long-unsigned</div><div>[6] ,</div></div><div><div>octet-string</div><div>[9] ,</div></div><div><div>visible-string</div><div>[10] ,</div></div><div><div>UTF8-string</div><div>[12] ,</div></div><div><div>integer</div><div>[15] ,</div></div><div><div>long</div><div>[16] ,</div></div><div><div>unsigned</div><div>[17] ,</div></div><div><div>long-unsigned</div><div>[18] ,</div></div><div><div>long64</div><div>[20] ,</div></div><div><div>long64-unsigned</div><div>[21] ,</div></div><div><div>enum</div><div>[22] ,</div></div><div><div>float32</div><div>[23] ,</div></div><div><div>float64</div><div>[24] ,</div></div><div><div>date_time</div><div>[25] ,</div></div><div><div>date</div><div>[26] ,</div></div></div> <div>}</div>	<div>见 6.3.2</div>



表 25（续）

数据类型定义	说明
<div>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] , RCSD [96] }</div>	见 6.3.2

6.3.3.2 浮点数据类型

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

6.3.3.3 PIID 数据类型

PIID 数据类型定义及说明见表 26。

表 26 PIID 数据类型定义及说明

数据类型定义	说 明
PIID::=unsigned	PIID 用于客户机 APDU（Client-APDU）的各服务数据类型中，基本定义如下（具体应用约定应根据实际系统要求而定）： bit7（服务优先级）——0，一般；1，高级。在.response APDU 中，其值与对应.request APDU 中的相等。 bit6（保留）。 bit0～bit5（服务序号）——二进制编码表示 0～63，在.response APDU 中，其值与对应.request APDU 中的相等

6.3.3.4 PIID-ACD 数据类型

PIID-ACD 数据类型定义及说明见表 27。

表 27 PIID-ACD 数据类型定义及说明

数据类型定义	说 明
PIID-ACD::=unsigned	PIID-ACD 用于服务器 APDU（Server-APDU）的各服务数据类型中，基本定义如下（具体应用约定应根据实际系统要求而定）： bit7（服务优先级）——0，一般；1，高级。在.response APDU 中，其值与对应.request APDU 中的相等。 bit6（请求访问 ACD）——0，不请求；1，请求。 bit0～bit5（服务序号）——二进制编码表示 0～63，在.response APDU 中，其值与对应.request APDU 中的相等

## 6.3.3.5 OAD 数据类型

OAD 数据类型定义及说明见表 28。

表 28 OAD 数据类型定义及说明

数据类型定义	说 明
<pre> 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) } </pre>	<p>对象标识——见 7.4</p> <p>对象属性标识及其特征——用 bit0~bit7 表示 8 位位组的最低位到最高位，其中：</p> <p>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；</p> <p>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。</p> <p>属性内元素索引——00H 表示整个属性全部内容。如果属性是一个结构或数组，01H 指向对象属性的第一个元素；如果属性是一个记录型的存储区，非零值 <math>n</math> 表示最近第 <math>n</math> 次的记录</p>

## 6.3.3.6 ROAD 数据类型

ROAD 数据类型定义及说明见表 29。

表 29 ROAD 数据类型定义及说明

数据类型定义	说 明
<pre> ROAD ::= SEQUENCE {     对象属性描述符      OAD,     关联对象属性描述符  SEQUENCE OF OAD } </pre>	<p>ROAD 用于描述记录型对象中的一个或若干个关联对象属性。</p> <p>OAD——见 6.3.3.5</p>

## 6.3.3.7 RSD 数据类型

RSD 数据类型定义及说明见表 30。

表 30 RSD 数据类型定义及说明

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>	<p>RSD 用于选择记录型对象属性的各条记录，即二维记录表的行选择，其通过对构成记录的某些对象属性数值进行指定来进行选择，范围选择区间为前闭后开，即〔起始值，结束值）。</p> <p>例如：事件类对象的事件记录表属性、冻结类对象的冻结数据记录表属性、采集监控类的采集数据记录表。</p> <p>应用提示：</p> <p>1) 对于事件记录，通常使用事件发生时间进行选择；</p> <p>2) 对于冻结数据记录，通常使用冻结时间进行选择</p>



表 30 (续)

数据类型定义		说 明
Selector1::=SEQUENCE { 对象属性描述符      OAD, 数值                      Data }		Selector1 为指定对象指定值
Selector2::=SEQUENCE { 对象属性描述符      OAD, 起始值                Data, 结束值                Data, 数据间隔              Data }		Selector2 为指定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；其他场景下通常间隔数据类型与数值类型相同
Selector3::=SEQUENCE OF Selector2		Selector3 为组合筛选，即若干个指定对象连续值
Selector4::=SEQUENCE { 采集启动时间          date_time_s, 表计集合              MS }		Selector4 为指定表计集合、指定采集启动时间。 MS——见 6.3.3.10
Selector5::=SEQUENCE { 采集存储时间          date_time_s, 表计集合              MS }		Selector5 为指定表计集合、指定采集存储时间。 MS——见 6.3.3.10
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 6.3.3.10
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 6.3.3.10
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、指定采集到时间区间内连续间隔值。 MS——见 6.3.3.10
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 6.3.3.10

## 6.3.3.8 RCSD 数据类型

RCSD 数据类型定义及说明见表 31。

表 31 RCSD 数据类型定义及说明

数据类型定义	说 明
RCSD ::= SEQUENCE OF CSD	<p>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。例如：事件记录或冻结数据记录中的某关联对象属性数据列。</p> <p>当无 OAD 时，RCSD=0，即 SEQUENCE OF 的数据项个数为 0，表示“不选择（即全选）”</p>

## 6.3.3.9 CSD 数据类型

CSD 数据类型定义及说明见表 32。

表 32 CSD 数据类型定义及说明

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 } 	<p>CSD 用于描述记录型对象中记录的列关联对象属性。</p> <p>OAD——见 6.3.3.5。</p> <p>ROAD——见 6.3.3.6</p>

## 6.3.3.10 MS 数据类型

MS 数据类型定义及说明见表 33。

表 33 MS 数据类型定义及说明

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 } 	<p>无表计——相当于无效配置。</p> <p>全部用户地址——全部可采集的表计。</p> <p>一组用户类型——指定的若干用户类型的表计。</p> <p>一组用户地址——指定的若干电能表通信地址的表计。</p> <p>一组配置序号——指定的若干电能表配置序号的表计。</p> <p>一组用户类型区间——按数组形式给出用户类型范围。</p> <p>一组用户地址区间——按数组形式给出表计通信地址范围。</p> <p>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</p>

## 6.3.3.11 DAR 数据类型

DAR 数据类型定义及说明见表 34。

表 34 DAR 数据类型定义及说明

数据类型定义	说 明
<pre>DAR ::= ENUMERATED {     成功 (0),     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 P1P2 不正确 (34),     ESAM 的 LC 错误 (35),     其他 (255) }</pre>	DAR 采用枚举方式来描述数据访问的各种可能结果

6.3.3.12 OMD 数据类型

OMD 数据类型定义及说明见表 35。

表 35 OMD 数据类型定义及说明

数据类型定义	说 明
<pre>OMD ::= SEQUENCE {     对象标识 OI,     方法标识 unsigned (1...255),     操作模式 unsigned (0) }</pre>	OMD 用于描对象的方法。 OI——见 7.4。 方法标识——即对象方法编号。 操作模式——值默认为 0

## 6.3.3.13 TSA 数据类型

TSA 数据类型定义及说明见表 36。

表 36 TSA 数据类型定义及说明

数据类型定义	说 明
TSA::=octet-string (SIZE (2..17) )	见 5.1.4.2.1

## 6.3.3.14 Scaler\_Unit 数据类型

Scaler\_Unit 数据类型定义及说明见表 37。

表 37 Scaler\_Unit 数据类型定义及说明

数据类型定义	说 明
Scaler_Unit::=SEQUENCE { 换算 integer, 单位 ENUMERATED }	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。 单位——枚举类型定义物理单位，详见附录 B

## 6.3.3.15 MAC 数据类型

MAC 数据类型定义及说明见表 38。

表 38 MAC 数据类型定义及说明

数据类型定义	说 明
MAC::=octet-string	MAC 见 3.1.14

## 6.3.3.16 SID 数据类型

SID 数据类型定义及说明见表 39。

表 39 SID 数据类型定义及说明

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

## 6.3.3.17 SID\_MAC 标识类型

SID\_MAC 数据类型定义及说明见表 40。

表 40 SID\_MAC 数据类型定义及说明

数据类型定义	说 明
SID_MAC::=SEQUENCE { 安全标识 SID, 数据 MAC MAC }	SID_MAC 为 ESAM 所属安全标识以及消息鉴别码

6.3.3.18 RN 数据类型

RN 数据类型定义及其说明见表 41。

表 41 RN 数据类型定义及说明

数据类型定义	说 明
RN ::= octet-string	RN 为 ESAM 生成用于加密的信息串

6.3.3.19 ConnectMechanismInfo 数据类型

ConnectMechanismInfo 数据类型定义及说明见表 42。

表 42 ConnectMechanismInfo 数据类型定义及说明

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 }	ConnectMechanismInfo 为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 } SignatureSecurity ::= SEQUENCE { 密文 2           octet-string, 客户机签名 2   octet-string }	密文 1——对客户机产生的随机数加密得到的密文。 客户机签名 1——客户机对密文 1 的签名。 密文 2——客户机（主站）对服务器（终端）产生的主站证书等 数据加密信息。 客户机签名 2——客户机对密文 2 的签名

6.3.3.20 ConnectResult 数据类型

ConnectResult 数据类型定义及说明见表 43。

表 43 ConnectResult 数据类型定义及说明

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

6.3.3.21 **ConnectResponseInfo** 数据类型

ConnectResponseInfo 数据类型定义及说明见表 44。

表 44 **ConnectResponseInfo** 数据类型定义及说明

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

6.3.3.22 **TI** 数据类型

TI 数据类型定义及说明见表 45。

表 45 **TI** 数据类型定义及说明

数据类型定义	说 明
<pre>TI ::= SEQUENCE {     单位  ENUMERATED     {         秒      (0),         分      (1),         时      (2),         日      (3),         月      (4),         年      (5)     },     间隔值 long-unsigned }</pre>	TI 用于表示时间间隔的间隔值及其时间单位； 间隔值——为 0 时表示无间隔

6.3.3.23 **Region** 数据类型

Region 数据类型定义及说明见表 46。

表 46 **Region** 数据类型定义及说明

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]

6.3.3.24 OI 数据类型

OI 数据类型定义及说明见表 47。

表 47 OI 数据类型定义及说明

数据类型定义	说 明
OI ::= long-unsigned	OI 见 7.4

6.3.3.25 date\_time 数据类型

date\_time 数据类型定义及说明见表 48。

表 48 date\_time 数据类型定义及说明

数据类型定义	说 明
<pre>date_time ::= SEQUENCE {     year          long-unsigned,     month         unsigned,     day_of_month  unsigned,     day_of_week   unsigned,     hour          unsigned,     minute        unsigned,     second        unsigned,     millisecond    long-unsigned }</pre>	<p>date_time 为日期和时间的十六进制格式。</p> <p>year、millisecond——FFFFH 时表示无效。</p> <p>month、day_of_month、day_of_week、hour、minute、second——FFH 时表示无效。</p> <p>day_of_week——0 表示周日，1~6 分别表示周一~周六</p>

6.3.3.26 date\_time\_s 数据类型

date\_time\_s 数据类型定义及说明见表 49。

表 49 date\_time\_s 数据类型定义及说明

数据类型定义	说 明
<pre>date_time_s ::= SEQUENCE {     year          long-unsigned,     month         unsigned,     day           unsigned,     hour          unsigned,     minute        unsigned,     second        unsigned }</pre>	<p>date_time_s 为日期和时间的十六进制格式。</p> <p>year——FFFFH 时表示无效。</p> <p>month、day、hour、minute、second——FFH 时表示无效</p>

6.3.3.27 date 数据类型

date 数据类型定义及说明见表 50。

表 50   date 数据类型定义及说明

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>	date 为日期的十六进制格式。 year=FFFFH 表示无效。 month、day_of_month、day_of_week—— 为 FFH 时表示无效

6.3.3.28   time 数据类型

time 数据类型定义及说明见表 51。

表 51   time 数据类型定义及说明

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

6.3.3.29   COMDCB 数据类型

COMDCB 数据类型定义及说明见表 52。

表 52   COMDCB 数据类型定义及说明

数据类型定义	说明
<pre>COMDCB ::= SEQUENCE {     波特率      ENUMERATED     {         300bit/s (0),    600bit/s (1),    1200bit/s (2),         2400bit/s (3),    4800bit/s (4),    7200bit/s (5),         9600bit/s (6),    19200bit/s (7),    38400bit/s (8),         57600bit/s (9),    115200bit/s (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)} }</pre>	

6.3.4   应用层数据单元

6.3.4.1   预连接数据单元

LINK-APDU 定义及说明见表 53。



表 53 LINK-APDU 定义及说明

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>	

6.3.4.2 客户机应用层协议数据单元

Client-APDU 定义及说明见表 54。

表 54 Client-APDU 定义及说明

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,         异常响应          [110]        ERROR-Response     },     时间标签 TimeTag OPTIONAL }</pre>	

6.3.4.3 服务器应用层协议数据单元

Server-APDU 定义及说明见表 55。

表 55 Server-APDU 定义及说明

数据类型定义	说明
<pre>Server-APDU ::= SEQUENCE {     应用层服务 CHOICE     {         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,         异常响应          [238]        ERROR-Response     },     跟随上报信息域 FollowReport OPTIONAL,     时间标签 TimeTag OPTIONAL }</pre>	

## 6.3.4.4 安全传输协议数据单元

SECURITY-APDU 定义及说明见表 56。

表 56 Security-APDU 定义及说明

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

## 6.3.5 预连接

## 6.3.5.1 LINK-Request 数据类型

LINK-Request 数据类型定义及说明见表 57。

表 57 LINK-Request 数据类型定义及说明

数据类型定义	说明
<pre>LINK-Request ::= SEQUENCE {     服务序号-优先级-ACD    PIID-ACD     请求类型    ENUMERATED     {         登录      (0),         心跳      (1),         退出登录  (2)     },     心跳周期    long-unsigned,     请求时间    date_time }</pre>	<p>PIID-ACD——见 6.3.3.4。 心跳周期——单位：s。 date_time——见 6.3.3.25</p>

## 6.3.5.2 LINK-Response 数据类型

LINK-Response 数据类型定义及说明见表 58。

表 58 LINK-Response 数据类型定义及说明

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 6.3.3.3。 date_time——见 6.3.3.25</p>

表 58 (续)

数据类型定义	说明
<pre>Result ::= bit-string (SIZE (8)) {     时钟可信标识      (0),     保留 bit6          (1),     保留 bit5          (2),     保留 bit4          (3),     保留 bit3          (4),     结果 bit2          (5),     结果 bit1          (6),     结果 bit0          (7) }</pre>	<p>时钟可信标识——用于表示响应方的时钟是否可信（准确）：0，不可信；1，可信。</p> <p>结果 bit0～结果 bit2——二进制编码表示：0，成功；1，地址重复；2，非法设备；3，容量不足；其他值：保留</p>

6.3.6 建立应用连接

6.3.6.1 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。

6.3.6.2 CONNECT-Request 数据类型

CONNECT-Request 数据类型定义及说明见表 59。

表 59 CONNECT-Request 数据类型定义及说明

数据类型定义	说明
<pre>Connect-Request ::= SEQUENCE {     服务序号-优先级      PIID,     期望的应用层协议版本号 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 APDU 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 }</pre>	<p>PIID——见 6.3.3.3。</p> <p>ProtocolConformance——定义见附录 C。</p> <p>FunctionConformance——定义见附录 C。</p> <p>客户机发送帧最大尺寸——单位：字节。</p> <p>客户机接收帧最大尺寸——单位：字节。</p> <p>客户机接收帧最大窗口尺寸——单位：个。</p> <p>期望的应用连接超时时间——单位：s。</p>

6.3.6.3 CONNECT-Response 数据类型

CONNECT-Response 数据类型定义及说明见表 60。

表 60 CONNECT-Response 数据类型定义及说明

数据类型定义	说明
<pre>CONNECT-Response ::= SEQUENCE {     服务序号-优先级-ACD   PIID-ACD,     服务器厂商版本信息     FactoryVersion,     商定的应用层协议版本号 long-unsigned,     商定的协议一致性块     ProtocolConformance, }</pre>	<p>PIID-ACD——见 6.3.3.4。</p> <p>ProtocolConformance——定义见附录 C。</p> <p>FunctionConformance——定义见附录 C。</p> <p>服务器发送帧最大尺寸——单位：字节。</p> <p>服务器接收帧最大尺寸——单位：字节。</p> <p>服务器接收帧最大窗口尺寸——单位：个</p>

表 60（续）

数据类型定义		说明
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 APDU 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	
}		
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) ) ,	
厂家扩展信息	visible-string (SIZE (8) )	
}		

6.3.7 断开应用连接

6.3.7.1 RELEASE-Request 数据类型

RELEASE-Request 数据类型定义及说明见表 61。

表 61 RELEASE-Request 数据类型定义及说明

数据类型定义	说明
RELEASE-Request ::= SEQUENCE { 服务序号-优先级 PIID }	PIID——见 6.3.3.3

6.3.7.2 RELEASE-Response 数据类型

RELEASE-Response 数据类型定义及说明见表 62。

表 62 RELEASE-Response 数据类型定义及说明

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 6.3.3.4

6.3.7.3 RELEASE-Notification 数据类型

RELEASE-Notification 数据类型定义及说明见表 63。

表 63 RELEASE-Notification 数据类型定义及说明

数据类型定义	说明
<pre>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 }</pre>	PIID-ACD——见 6.3.3.4 date_time_s——见 6.3.3.26

6.3.8 读取

6.3.8.1 GET-Request 数据类型

GET-Request 数据类型定义及说明见表 64。

表 64 GET-Request 数据类型定义及说明

数据类型定义	说明
<pre>GET-Request ::= CHOICE {     读取一个对象属性请求      [1] GetRequestNormal,     读取若干个对象属性请求    [2] GetRequestNormalList,     读取一个记录型对象属性请求 [3] GetRequestRecord,     读取若干个记录型对象属性请求 [4] GetRequestRecordList,     读取分帧响应的下一个数据块请求 [5] GetRequestNext     读取一个对象属性的 MD5 值    [6] GetRequestMD5 }</pre>	

6.3.8.1.1 GetRequestNormal 数据类型

GetRequestNormal 数据类型定义及说明见表 65。

表 65 GetRequestNormal 数据类型定义及说明

数据类型定义	说明
<pre>GetRequestNormal ::= SEQUENCE {     服务序号-优先级      PIID,     一个对象属性描述符  OAD }</pre>	PIID——见 6.3.3.3。 OAD——见 6.3.3.5

6.3.8.1.2 GetRequestNormalList 数据类型

GetRequestNormalList 数据类型定义及说明见表 66。

表 66 GetRequestNormalList 数据类型定义及说明

数据类型定义	说明
<pre> GetRequestNormalList ::= SEQUENCE {     服务序号-优先级          PIID,     若干个对象属性描述符    SEQUENCE OF OAD } </pre>	<p>PIID——见 6.3.3.3。  OAD——见 6.3.3.5</p>

## 6.3.8.1.3 GetRequestRecord 数据类型

GetRequestRecord 数据类型定义及说明见表 67。

表 67 GetRequestRecord 数据类型定义及说明

数据类型定义	说明
<pre> GetRequestRecord ::= SEQUENCE {     服务序号-优先级          PIID,     读取一个记录型对象属性  GetRecord } </pre>	<p>PIID——见 6.3.3.3</p>
<pre> GetRecord ::= 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        RCSD } </pre>	<p>OAD——见 6.3.3.5。  RSD——见 6.3.3.7。  RCSD——见 6.3.3.8</p>

## 6.3.8.1.4 GetRequestRecordList 数据类型

GetRequestRecordList 数据类型定义及说明见表 68。

表 68 GetRequestRecordList 数据类型定义及说明

数据类型定义	说明
<pre> 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 } </pre>	<p>PIID——见 6.3.3.3。  GetRecord——见表 67</p>

## 6.3.8.1.5 GetRequestNext 数据类型

GetRequestNext 数据类型定义及说明见表 69。

表 69 GetRequestNext 数据类型定义及说明

数据类型定义	说明
<pre> GetRequestNext ::= SEQUENCE {     服务序号-优先级          PIID,     正确接收的最近一次数据块序号 long-unsigned } </pre>	<p>PIID——见 6.3.3.3</p>

6.3.8.1.6 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 70。

表 70 GetRequestMD5 数据类型定义及说明

数据类型定义	说明
<pre>GetRequestMD5 ::= SEQUENCE {     服务序号-优先级      PIID,     一个对象属性描述符  OAD }</pre>	PIID——见 6.3.3.3

6.3.8.2 GET-Response 数据类型

GET-Response 数据类型定义及说明见表 71。

表 71 GET-Response 数据类型定义及说明

数据类型定义	说明
<pre>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,     分帧响应一个数据块          [5]  GetResponseNext     读取一个对象属性的 MD5 值的响应 [6]  GetResponseMD5 }</pre>	

6.3.8.2.1 GetResponseNormal 数据类型

GetResponseNormal 数据类型定义及说明见表 72。

表 72 GetResponseNormal 数据类型定义及说明

数据类型定义	说明
<pre>GetResponseNormal ::= SEQUENCE {     服务序号-优先级-ACD  PIID-ACD,     一个对象属性及其结果  A-ResultNormal }</pre>	PIID-ACD——见 6.3.3.4
<pre>A-ResultNormal ::= SEQUENCE {     对象属性描述符  OAD,     及其结果        Get-Result }</pre>	OAD——见 6.3.3.5。
<pre>Get-Result ::= CHOICE {     错误信息  [0]  DAR,     数据      [1]  Data }</pre>	DAR——见 6.3.3.11 Data——见 6.3.3.1

6.3.8.2.2 **GetResponseNormalList** 数据类型

GetResponseNormalList 数据类型定义及说明见表 73。

表 73 **GetResponseNormalList** 数据类型定义及说明

数据类型定义	说明
<pre>GetResponseNormalList ::= SEQUENCE {     服务序号-优先级-ACD      PIID-ACD,     若干个对象属性及其结果  SEQUENCE OF A-ResultNormal }</pre>	<p>PIID-ACD——见 6.3.3.4。 A-ResultNormal——见表 72</p>

6.3.8.2.3 **GetResponseRecord** 数据类型

GetResponseRecord 数据类型定义及说明见表 74。

表 74 **GetResponseRecord** 数据类型定义及说明

数据类型定义	说明
<pre>GetResponseRecord ::= SEQUENCE {     服务序号-优先级-ACD      PIID-ACD,     一个记录型对象属性及其结果 A-ResultRecord }</pre>	<p>PIID-ACD——见 6.3.3.4</p>
<pre>A-ResultRecord ::= SEQUENCE {     记录型对象属性描述符      OAD,     一行记录 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     } }</pre>	<p>OAD——见 6.3.3.5。 一行记录 N 列属性描述符——即记录表的表头信息。</p>
<pre>A-RecordRow ::= SEQUENCE {     第 1 列数据  Data,     第 2 列数据  Data,     ...     第 N 列数据  Data }</pre>	<p>第 1 列~第 N 列——其排列顺序与“一行记录 N 列属性描述符”的排列顺序一致。 Data——见 6.3.3.1</p>

6.3.8.2.4 **GetResponseRecordList** 数据类型

GetResponseRecordList 数据类型定义及说明见表 75。

表 75 **GetResponseRecordList** 数据类型定义及说明

数据类型定义	说明
<pre>GetResponseRecordList ::= SEQUENCE {     服务序号-优先级-ACD      PIID-ACD,     若干个记录型对象属性及其结果 SEQUENCE OF A-ResultRecord }</pre>	<p>PIID-ACD——见 6.3.3.4。 OAD——见 6.3.3.5。 A-ResultRecord——见表 74</p>



6.3.8.2.5 **GetResponseNext** 数据类型

GetResponseNext 数据类型定义及说明见表 76。

表 76 **GetResponseNext** 数据类型定义及说明

数据类型定义	说明
<pre>GetResponseNext ::= SEQUENCE {     服务序号-优先级-ACD      PIID-ACD,     末帧标识                  BOOLEAN,     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     } }</pre>	<p>PIID-ACD——见 6.3.3.4。  DAR——见 6.3.3.11。  A-ResultNormal——见表 72。  A-ResultRecord——见表 74</p>

6.3.8.2.6 **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 77。

表 77 **GetResponseMD5** 数据类型定义及说明

数据类型定义	说明
<pre>GetResponseMD5 ::= SEQUENCE {     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     } }</pre>	<p>PIID-ACD——见 6.3.3.4。  DAR——见 6.3.3.11。  MD5 值——使用编码后的对象属性值进行计算</p>

6.3.9 设置

6.3.9.1 **SET-Request** 数据类型

SET-Request 数据类型定义及说明见表 78。

表 78 **SET-Request** 数据类型定义及说明

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

## 6.3.9.1.1 SetRequestNormal 数据类型

SetRequestNormal 数据类型定义及说明见表 79。

表 79 SetRequestNormal 数据类型定义及说明

数据类型定义	说明
<pre>SetRequestNormal ::= SEQUENCE {     服务序号-优先级      PIID,     一个对象属性描述符  OAD,     数据                  Data }</pre>	<p>PIID——见 6.3.3.3。  OAD——见 6.3.3.5。  Data——见 6.3.3.1</p>

## 6.3.9.1.2 SetRequestNormalList 数据类型

SetRequestNormalList 数据类型定义及说明见表 80。

表 80 SetRequestNormalList 数据类型定义及说明

数据类型定义	说明
<pre>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     } }</pre>	<p>PIID——见 6.3.3.3。  OAD——见 6.3.3.5。  Data——见 6.3.3.1</p>

## 6.3.9.1.3 SetThenGetRequestNormalList 数据类型

SetThenGetRequestNormalList 数据类型定义及说明见表 81。

表 81 SetThenGetRequestNormalList 数据类型定义及说明

数据类型定义	说明
<pre>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,         一个读取的对象属性  OAD,         延时读取时间          unsigned     } }</pre>	<p>PIID——见 6.3.3.3。  OAD——见 6.3.3.5。  Data——见 6.3.3.1。  延时读取时间——单位：s，0 表示采用服务器默认的延时时间</p>

## 6.3.9.2 SET-Response 数据类型

SET-Response 数据类型定义及说明见表 82。

表 82 SET-Response 数据类型定义及说明

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

6.3.9.2.1 SetResponseNormal 数据类型

SetResponseNormal 数据类型定义及说明见表 83。

表 83 SetResponseNormal 数据类型定义及说明

数据类型定义	说明
<pre>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 }</pre>	PIID-ACD——见 6.3.3.4。 OAD——见 6.3.3.5。 DAR——见 6.3.3.11

6.3.9.2.2 SetResponseNormalList 数据类型

SetResponseNormalList 数据类型定义及说明见表 84。

表 84 SetResponseNormalList 数据类型定义及说明

数据类型定义	说明
<pre>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   {     一个对象属性描述符    OAD,     设置执行结果          DAR   } }</pre>	PIID-ACD——见 6.3.3.4。 OAD——见 6.3.3.5。 DAR——见 6.3.3.11

6.3.9.2.3 SetThenGetResponseNormalList 数据类型

SetThenGetResponseNormalList 数据类型定义及说明见表 85。

表 85 SetThenGetResponseNormalList 数据类型定义及说明

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>	PIID-ACD——见 6.3.3.4。 OAD——见 6.3.3.5。 DAR——见 6.3.3.11。 A-ResultNormal——见表 72

## 6.3.10 操作

## 6.3.10.1 ACTION-Request 数据类型

## 6.3.10.1.1 ACTION-Request 数据类型定义

ACTION-Request 数据类型定义及说明见表 86。

表 86 ACTION-Request 数据类型定义及说明

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

## 6.3.10.1.2 ActionRequest 数据类型

ActionRequest 数据类型定义及说明见表 87。

表 87 ActionRequest 数据类型定义及说明

数据类型定义	说明
<pre> ActionRequest ::= 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>	<p>PIID——见 6.3.3.3。          OMD——见 6.3.3.12。          Data——见 6.3.3.1</p>

## 6.3.10.1.3 ActionRequestList 数据类型

ActionRequestList 数据类型定义及说明见表 88。

表 88 ActionRequestList 数据类型定义及说明

数据类型定义	说明
<pre> ActionRequestList ::= SEQUENCE {     服务序号-优先级    PIID,     若干个对象属性     SEQUENCE OF     {         一个对象方法描述符 OMD,         方法参数           Data     } } </pre>	<p>PIID——见 6.3.3.3。          OMD——见 6.3.3.12。          Data——见 6.3.3.1</p>

## 6.3.10.1.4 ActionThenGetRequestNormalList 数据类型

ActionThenGetRequestNormalList 数据类型定义及说明见表 89。

表 89 ActionThenGetRequestNormalList 数据类型定义及说明

数据类型定义	说明
<pre>ActionThenGetRequestNormalList ::= SEQUENCE {     服务序号-优先级 PIID,     若干个操作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 6.3.3.3。 OAD——见 6.3.3.5。 OMD——见 6.3.3.12。 Data——见 6.3.3.1。 读取延时——单位：s。0 表示取服务器默认的延时时间

6.3.10.2 ACTION-Response 数据类型

6.3.10.2.1 ACTION-Response 数据类型定义

ACTION-Response 数据类型定义及说明见表 90。

表 90 ACTION-Response 数据类型定义及说明

数据类型定义	说明
<pre>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 }</pre>	

6.3.10.2.2 ActionResponseNormal 数据类型

ActionResponseNormal 数据类型定义及说明见表 91。

表 91 ActionResponseNormal 数据类型定义及说明

数据类型定义	说明
<pre>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 }</pre>	PIID-ACD——见 6.3.3.4。 OMD——见 6.3.3.12。 DAR——见 6.3.3.11。 Data——见 6.3.3.1。

6.3.10.2.3 ActionResponseNormalList 数据类型

ActionResponseNormalList 数据类型定义及说明见表 92。

表 92 ActionResponseNormalList 数据类型定义及说明

数据类型定义	说明
<pre>ActionResponseNormalList ::= SEQUENCE {</pre>	PIID-ACD——见 6.3.3.4。 OMD——见 6.3.3.12。

表 92 (续)

数据类型定义	说明
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 } }	DAR——见 6.3.3.11。 Data——见 6.3.3.1

#### 6.3.10.2.4 ActionThenGetResponseNormalList 数据类型

ActionThenGetResponseNormalList 数据类型定义及说明见表 93。

表 93 ActionThenGetResponseNormalList 数据类型定义及说明

数据类型定义	说明
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, 一个对象及结果                      A-ResultNormal } }	PIID-ACD——见 6.3.3.4。 OMD——见 6.3.3.12。 DAR——见 6.3.3.11。 OAD——见 6.3.3.5。 Data——见 6.3.3.1。 A-ResultNormal——见表 72

### 6.3.11 上报

#### 6.3.11.1 REPORT-Notification 数据类型

##### 6.3.11.1.1 REPORT-Notification 数据类型定义

REPORT-Notification 数据类型定义及说明见表 94。

表 94 REPORT-Notification 数据类型定义及说明

数据类型定义	说明
REPORT-Notification ::= CHOICE { 上报若干个对象属性              [1] ReportNotificationList, 上报若干个记录型对象属性        [2] ReportNotificationRecordList 上报透明数据                      [3] ReportNotificationTransData }	

##### 6.3.11.1.2 ReportNotificationList 数据类型

ReportNotificationList 数据类型定义及说明见表 95。

表 95 ReportNotificationList 数据类型定义及说明

数据类型定义	说明
ReportNotificationList ::= SEQUENCE { 服务序号-优先级-ACD       PIID-ACD, 若干个对象属性及其数据   SEQUENCE OF A-ResultNormal }	PIID-ACD——见 6.3.3.4。 A-ResultNormal——见表 72

6.3.11.1.3 ReportNotificationRecordList 数据类型

ReportNotificationRecordList 数据类型定义及说明见表 96。

表 96 ReportNotificationRecordList 数据类型定义及说明

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其数据   SEQUENCE OF A-ResultRecord }	PIID-ACD——见 6.3.3.4。 A-ResultRecord——见表 74

6.3.11.1.4 ReportNotificationTransData 数据类型

通知上报透明数据（ReportNotificationTransData）的数据类型定义见表 97。

表 97 ReportNotificationTransData 数据类型定义及说明

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 6.3.3.4

6.3.11.2 REPORT-Response 数据类型

6.3.11.2.1 REPORT-Response 数据类型

REPORT-Response 数据类型定义及说明见表 98。

表 98 REPORT-Response 数据类型定义及说明

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 }	

6.3.11.2.2 ReportResponseList 数据类型

ReportResponseList 数据类型定义及说明见表 99。

表 99 ReportResponseList 数据类型定义及说明

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，为接收到的记录型对象属性数值的对象属性描述符，用于向服务器表明接收确认。 PIID-ACD——见 6.3.3.4。 OAD——见 6.3.3.5

#### 6.3.11.2.3 ReportResponseRecordList 数据类型

ReportResponseRecordList 数据类型定义及说明见表 100。

表 100 ReportResponseRecordList 数据类型定义及说明

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——为接收到的记录型对象属性数值的对象属性描述符，用于向服务器表明接收确认。  PIID——见 6.3.3.3。 OAD——见 6.3.3.5

#### 6.3.11.2.4 ReportResponseTransData 数据类型

响应上报透明数据（ReportResponseTransData）的数据类型定义及说明见表 101。

表 101 ReportResponseTransData 数据类型定义及说明

数据类型定义	说明
ReportResponseTransData ::= SEQUENCE { 服务序号-优先级     PIID }	PIID——见 6.3.3.3

### 6.3.12 代理

#### 6.3.12.1 PROXY-Request 数据类型

#### 6.3.12.1.1 PROXY-Request 数据类型

PROXY-Request 数据类型定义及说明见表 102。

表 102 PROXY-Request 数据类型定义及说明

数据类型定义		说明
PROXY-Request ::= CHOICE { 代理读取若干个服务器的若干个对象属性请求 代理读取一个服务器的一个记录型对象属性请求           }	[1] ProxyGetRequestList, [2] ProxyGetRequestRecord,	



表 102（续）

数据类型定义	说明
代理设置若干个服务器的若干个对象属性请求 [3] ProxySetRequestList, 代理设置后读取若干个服务器的若干个对象属性请求 [4] ProxySetThenGetRequestList, 代理操作若干个服务器的若干个对象方法请求 [5] ProxyActionRequestList, 代理操作后读取若干个服务器的若干个对象方法和属性请求 [6] ProxyActionThenGetRequestList, 代理透明转发命令请求 [7] ProxyTransCommandRequest }	

6.3.12.1.2 ProxyGetRequestList 数据类型

ProxyGetRequestList 数据类型定义及说明见表 103。

表 103 ProxyGetRequestList 数据类型定义及说明

数据类型定义	说明
ProxyGetRequestList ::= SEQUENCE { 服务序号-优先级 PIID, 整个代理请求的超时时间 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 6.3.3.3。 TSA——见 6.3.3.13。 OAD——见 6.3.3.5。 代理一个服务器的超时时间——单位：s。0 表示由代理服务器确定超时时间。 整个代理请求的超时时间——单位：s；非 0 值

6.3.12.1.3 ProxyGetRequestRecord 数据类型

ProxyGetRequestRecord 数据类型定义及说明见表 104。

表 104 ProxyGetRequestRecord 数据类型定义及说明

数据类型定义	说明
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, 记录选择描述符 RSD, 记录列选择描述符 RCSD }	PIID——见 6.3.3.3。 TSA——见 6.3.3.13。 OAD——见 6.3.3.5。 RSD——见 6.3.3.7。 RCSD——见 6.3.3.8。 整个代理请求的超时时间——单位：s；非 0 值

6.3.12.1.4 ProxySetRequestList 数据类型

ProxySetRequestList 数据类型定义及说明见表 105。

表 105 ProxySetRequestList 数据类型定义及说明

数据类型定义	说明
ProxySetRequestList ::= SEQUENCE { 服务序号-优先级 PIID, }	PIID——见 6.3.3.3。 TSA——见 6.3.3.13。 OAD——见 6.3.3.5。

表 105 (续)

数据类型定义	说明
整个代理请求的超时时间 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, 代理一个服务器的超时时间 long-unsigned, 若干个对象属性描述符及其数据 SEQUENCE OF { 对象属性描述符 OAD, 及其数据 Data } } 	Data——见 6.3.3.1。 代理一个服务器的超时时间——单位: s。0 表示由代理服务器确定超时时间。 整个代理请求的超时时间——单位: s; 非 0 值

## 6.3.12.1.5 ProxySetThenGetRequestList 数据类型

ProxySetThenGetRequestList 数据类型定义及说明见表 106。

表 106 ProxySetThenGetRequestList 数据类型定义及说明

数据类型定义	说明
ProxySetThenGetRequestList ::= SEQUENCE { 服务序号-优先级 PIID, 整个代理请求的超时时间 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, 及其设置数值 Data, 读取的对象属性描述符 OAD, 及其延时读取时间 unsigned } } } 	PIID——见 6.3.3.3。 TSA——见 6.3.3.13。 OAD——见 6.3.3.5。 Data——见 6.3.3.1。 延时读取时间——单位: s。0 表示由目标服务器确定延时时间。 代理一个服务器的超时时间——单位: s。0 表示由代理服务器确定超时时间。 整个代理请求的超时时间——单位: s; 非 0 值

## 6.3.12.1.6 ProxyActionRequestList 数据类型

ProxyActionRequestList 数据类型定义及说明见表 107。

表 107 ProxyActionRequestList 数据类型定义及说明

数据类型定义	说明
ProxyActionRequestList ::= SEQUENCE { 服务序号-优先级 PIID, 整个代理请求的超时时间 long-unsigned, 代理若干个服务器的对象方法操作 SEQUENCE OF           }	PIID——见 6.3.3.3。 TSA——见 6.3.3.13。 OMD——见 6.3.3.12。 Data——见 6.3.3.1。

表 107（续）

数据类型定义	说明
<pre>{   一个目标服务器地址      TSA,   代理一个服务器的超时时间  long-unsigned,   若干个对象方法描述符及其参数 SEQUENCE OF   {     对象方法描述符  OMD,     及其方法参数    Data   } }</pre>	代理一个服务器的超时时间——单位：s；0 表示由代理服务器确定超时时间。 整个代理请求的超时时间——单位：s；非 0 值

6.3.12.1.7 ProxyActionThenGetRequestList 数据类型

ProxyActionThenGetRequestList 数据类型定义及说明见表 108。

表 108 ProxyActionThenGetRequestList 数据类型定义及说明

数据类型定义	说明
<pre>ProxyActionThenGetRequestList ::= SEQUENCE {   服务序号-优先级      PIID,   整个代理请求的超时时间  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,       及其方法参数          Data,       读取的对象属性描述符  OAD,       及其延时读取时间      unsigned     }   } }</pre>	PIID——见 6.3.3.3。 TSA——见 6.3.3.13。 OMD——见 6.3.3.12。 Data——见 6.3.3.1。 OAD——见 6.3.3.5。 延时读取时间——单位：s；0 表示采用服务器默认的延时时间。 代理一个服务器的超时时间——单位：s；0 表示由代理服务器确定超时时间。 整个代理请求的超时时间——单位：s；非 0 值

6.3.12.1.8 ProxyTransCommandRequest 数据类型

ProxyTransCommandRequest 数据类型定义及说明见表 109。

表 109 ProxyTransCommandRequest 数据类型定义及说明

数据类型定义	说明
<pre>ProxyTransCommandRequest ::= SEQUENCE {   服务序号-优先级      PIID,   数据转发端口          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,   透明转发命令          octet-string }</pre>	PIID——见 6.3.3.3。 OAD——见 6.3.3.5。 COMDCB——见 6.3.3.29

### 6.3.12.2 PROXY-Response 数据类型

#### 6.3.12.2.1 PROXY-Response 数据类型

PROXY-Response 数据类型定义及说明见表 110。

表 110 PROXY-Response 数据类型定义及说明

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响应     代理读取一个服务器的一个记录型对象属性响应     代理设置若干个服务器的若干个对象属性响应     代理设置后读取若干个服务器的若干个对象属性响应     代理操作若干个服务器的若干个对象方法响应     代理操作后读取若干个服务器的若干个对象方法和属性响应     代理透明转发命令响应 } </pre>	<p>[1] ProxyGetResponseList, [2] ProxyGetResponseRecord, [3] ProxySetResponseList, [4] ProxySetThenGetResponseList, [5] ProxyActionResponseList, [6] ProxyActionThenGetResponseList, [7] ProxyTransCommandResponse</p>

#### 6.3.12.2.2 ProxyGetResponseList 数据类型

ProxyGetResponseList 数据类型定义及说明见表 111。

表 111 ProxyGetResponseList 数据类型定义及说明

数据类型定义	说明
<pre> ProxyGetResponseList ::= 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,         若干个对象属性及结果  SEQUENCE OF A-ResultNormal     } } </pre>	<p>PIID-ACD——见 6.3.3.4。 TSA——见 6.3.3.13。 OAD——见 6.3.3.5。 A-ResultNormal——见表 72</p>

#### 6.3.12.2.3 ProxyGetResponseRecord 数据类型

ProxyGetResponseRecord 数据类型定义及说明见表 112。

表 112 ProxyGetResponseRecord 数据类型定义及说明

数据类型定义	说明
<pre> 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其结果  A-ResultRecord } </pre>	<p>PIID-ACD——见 6.3.3.4。 TSA——见 6.3.3.13。 A-ResultRecord——见表 74</p>

#### 6.3.12.2.4 ProxySetResponseList 数据类型

ProxySetResponseList 数据类型定义及说明见表 113。

表 113 ProxySetResponseList 数据类型定义及说明

数据类型定义	说明
<pre>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,         若干个对象属性描述符及其结果  SEQUENCE OF         {             对象属性描述符    OAD,             及其设置结果        DAR         }     } }</pre>	<p>PIID-ACD——见 6.3.3.4。  TSA——见 6.3.3.13。  OAD——见 6.3.3.5。  DAR——见 6.3.3.11</p>

6.3.12.2.5 ProxySetThenGetResponseList 数据类型

ProxySetThenGetResponseList 数据类型定义及说明见表 114。

表 114 ProxySetThenGetResponseList 数据类型定义及说明

数据类型定义	说明
<pre>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,             及其设置结果                DAR,             一个对象属性及结果        A-ResultNormal         }     } }</pre>	<p>PIID-ACD——见 6.3.3.4。  TSA——见 6.3.3.13。  OAD——见 6.3.3.5。  DAR——见 6.3.3.11。  A-ResultNormal——见表 72</p>

6.3.12.2.6 ProxyActionResponseList 数据类型

ProxyActionResponseList 数据类型定义及说明见表 115。

表 115 ProxyActionResponseList 数据类型定义及说明

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其结果  SEQUENCE OF         {             对象方法描述符    OMD, </pre>	<p>PIID-ACD——见 6.3.3.4。  TSA——见 6.3.3.13。  OMD——见 6.3.3.12。  DAR——见 6.3.3.11。  Data——见 6.3.3.1</p>

表 115 (续)

数据类型定义	说明
及其操作结果 操作返回数据 } } }	

## 6.3.12.2.7 ProxyActionThenGetResponseList 数据类型

ProxyActionThenGetResponseList 数据类型定义及说明见表 116。

表 116 ProxyActionThenGetResponseList 数据类型定义及说明

数据类型定义	说明
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, 及其操作结果           DAR, 操作返回数据           Data OPTIONAL, 一个对象属性及结果    A-ResultNormal } } }	PIID-ACD——见 6.3.3.4。 TSA——见 6.3.3.13。 OMD——见 6.3.3.12。 DAR——见 6.3.3.11。 OAD——见 6.3.3.5。 Data——见 6.3.3.1。 A-ResultNormal——见表 72

## 6.3.12.2.8 ProxyTransCommandResponse 数据类型

ProxyTransCommandResponse 数据类型定义及说明见表 117。

表 117 ProxyTransCommandResponse 数据类型定义及说明

数据类型定义	说明
ProxyTransCommandResponse ::= SEQUENCE { 服务序号-优先级-ACD   PIID-ACD, 数据转发端口           OAD, 透明转发命令返回结果   TransResult }	PIID-ACD——见 6.3.3.4。 OAD——见 6.3.3.5
TransResult ::= CHOICE { 错误信息       [0]   DAR, 返回数据       [1]   octet-string }	DAR——见 6.3.3.11

## 6.3.13 安全传输

## 6.3.13.1 SECURITY-Request 数据类型

SECURITY-Request 数据类型定义及说明见表 118。

表 118 SECURITY-Request 数据类型定义及说明

数据类型定义	说明
<pre>SECURITY-Request ::= SEQUENCE {     应用数据单元 CHOICE     {         明文应用数据单元 [0]  octet-string,         密文应用数据单元 [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+数据 MAC [2]  RN_MAC,         安全标识 [3]  SID     } }</pre>	SID——见 6.3.3.16。 RN——见 6.3.3.18。 SID_MAC——见 6.3.3.17
<pre>RN_MAC ::= SEQUENCE {     随机数 RN,     数据 MAC MAC }</pre>	MAC——见 6.3.3.15

6.3.13.2 SECURITY-Response 数据类型

SECURITY-Response 数据类型定义及说明见表 119。

表 119 SECURITY-Response 数据类型定义及说明

数据类型定义	说明
<pre>SECURITY-Response ::= SEQUENCE {     应用数据单元 CHOICE     {         明文应用数据单元 [0]  octet-string,         密文应用数据单元 [1]  octet-string,         异常错误 [2]  DAR     },     数据验证信息 CHOICE OPTIONAL     {         数据 MAC [0]  MAC         数据验证码 [1]  SID_MAC     } }</pre>	DAR——见 6.3.3.11。 MAC——见 6.3.3.15

6.3.14 跟随上报信息域

FollowReport 数据类型定义及说明见表 120。

表 120 FollowReport 数据类型定义及说明

数据类型定义	说明
<pre>FollowReport ::= CHOICE {     若干个对象属性及其数据 [1]  SEQUENCE OF A-ResultNormal,     若干个记录型对象属性及其数据 [2]  SEQUENCE OF A-ResultRecord }</pre>	A-ResultNormal——见表 71。 A-ResultRecord——见表 73

6.3.15 时间标签域

TimeTag 数据类型的定义及说明见表 121。

表 121 TimeTag 数据类型定义及说明

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 6.3.3.26。 TI——见 6.3.3.22

6.3.16 异常响应

响应异常（ERROR-Response）的数据类型定义及说明见表 122。

表 122 ERROR-Response 数据类型定义及说明

数据类型定义	说明
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) } }	PIID——见 6.3.3.3

6.4 应用层编码规范

应用层数据单元（APDU）遵循 A-XDR 编码规则，详见 DL/T 790.6。APDU 编码举例见附录 D。

7 接口类与对象标识

7.1 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）。

7.2 接口类的表示方法

接口类采用表格形式描述，其中包含类名、属性和方法等。接口类说明模板见表 123。

表 123 接口类说明模板

类名，class_id= <i>n</i>		实例数
属性		数据类型
1. 逻辑名（OI）	（static）	octet-string



表 123（续）

类名，class_id= <i>n</i>		实例数
属性		数据类型
2. ....	(.....)	.....
3. ....	(.....)	.....
方法		必选/可选
1. ....		.....
2. ....		.....

其中：

- a) 类名：类的说明，如电能量类、最大需量类、功率类、冻结类等。
- b) 实例数：规定逻辑设备内类的范例号的范围。
- c) 类标识码：即 class\_id 数值范围 0~255。
- d) 属性：规定类的属性。静态（static）表示终端自身不能更改的属性，如配置参数；动态（dyn.）表示载有过程的属性，此属性是由终端自己刷新的。
- e) 数据类型：定义属性的数据类型。
- f) 方法：规定类的方法“Method Name（）”，这些方法应在“方法说明”中说明。
- g) 必选/可选：定义方法是必选的还是可选的。
- h) 逻辑名：对象标识（OI），标记接口类对象实例。

每个属性和方法都应有详细说明。属性说明是说明属性的数据类型（如果是复杂数据类型），方法说明是说明对象的每个方法和执行的行为特征。

7.3 接口类

7.3.1 电能量类

电能量类定义了电能量数据信息，定义见表 124。

表 124 电 能 量 类 定 义

电能量类，class_id=1		0~ <i>n</i>
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
3. 换算及单位	(static)	Scaler_Unit
4. 高精度总及费率电能量数组	(dyn.)	array
5. 高精度换算及单位	(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 125。

表 125 电 能 量 类 属 性 说 明

编号	属性	说明
1	逻辑名::=octet-string	
2	总及费率电能数组::=array 电能 电能: CHOICE { double-long-unsigned [6] , double-long [5] }	包含总及 $n$ 个费率的电能, 规定其第一个数组元素是总电能, 后面依次排列费率 1~ $n$ 电能
3	换算及单位::=Scaler_Unit	见 6.3.3.14
4	扩展精度总及费率电能数组::=array 扩展精度电能 扩展精度电能: CHOICE { long64-unsigned [21] , long64 [20] }	包含总及 $n$ 个费率的电能, 规定其第一个数组元素是总电能, 后面依次排列费率 1~ $n$ 电能
5	扩展精度换算及单位::=Scaler_Unit	见 6.3.3.14

电能类方法说明见表 126。

表 126 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

### 7.3.2 最大需量类

最大需量类定义了最大需量数据信息, 定义见表 127。

表 127 最大需量接口类定义

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 128。

表 128 最大需量接口类属性说明

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] }	包含总及 $n$ 个费率的最大需量, 规定其第一个数组元素是总最大需量, 后面依次排列费率 1~ $n$ 最大需量
3	换算及单位::=Scaler_Unit	见 6.3.3.14。 最大需量发生时间无换算单位

最大需量类方法说明见表 129。

表 129 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

7.3.3 分相变量类

分相变量类定义了电压、电流、相角等分相变量信息, 定义见表 130。

表 130 分相变量类定义

分相变量类, class_id=3		$0 \sim n$
属性		数据类型
1. 逻辑名	(static)	octet-string
2. 分相数值组	(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 131。

表 131 分相变量类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	分相数值组::=array 分相数值 分相数值::=instance-specific	数值组按 A 相、B 相、C 相顺序排列。另外，当接线方式为单相时，A、B、C 三相改为 A 相（某一相）
3	换算及单位::=Scaler_Unit	见 6.3.3.14

分相变量类方法说明见表 132。

表 132 分相变量接口类方法说明

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

### 7.3.4 功率类

功率类定义了功率、功率因数等数据信息，定义见表 133。

表 133 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 134。

表 134 功率类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	总及分相数值组::=array 总或分相数值 总或分相数值::=instance-specific	数值组按总、A 相、B 相、C 相顺序排列，当接线方式为单相时，为总、A 相（某一相）
3	换算及单位::=Scaler_Unit	见 6.3.3.14

功率类方法说明见表 135。

表 135 功率类方法说明

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

### 7.3.5 谐波变量类

谐波变量类定义了谐波变量数据信息，定义见表 136。

表 136 谐波变量接口类定义

谐波变量接口类，class_id=5		0~n
属性		数据类型
1. 逻辑名	(static)	octet-string
2. A 相 n 次数值组	(dyn.)	array
3. B 相 n 次数值组	(dyn.)	array
4. C 相 n 次数值组	(dyn.)	array
5. 谐波次数 n	(static)	unsigned
6. 换算及单位	(static)	Scaler_Unit
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 137。

表 137 谐波变量类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	A 相 n 次数值组::=array A 相各次数值 A 相各次数值::=instance-specific	包含 A 相谐波相关数值
3	B 相 n 次数值组::=array B 相各次数值 B 相各次数值::=instance-specific	包含 B 相谐波相关数值
4	C 相 n 次数值组::=array C 相各次数值 C 相各次数值::=instance-specific	包含 C 相谐波相关数值
5	谐波次数 n	表示谐波相关数值组中的最高谐波次数
6	换算及单位::=Scaler_Unit	见 6.3.3.14

谐波变量类方法说明见表 138。

表 138 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

### 7.3.6 数据变量类

数据变量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 139。

表 139 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 140。

表 140 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	包含过程值或与过程值单元相关的状态值，数据类型依据“逻辑名”决定的对象实例而定义
3	换算及单位::=Scaler_Unit	见 6.3.3.14

数据变量类方法说明见表 141。

表 141 数据变量类方法说明

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

### 7.3.7 事件对象类

事件对象类定义了配置、存储事件记录数据信息，定义见表 142。

表 142 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 143。

表 143 事件对象类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	<p>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,   ...   第 n 个关联对象属性的数据 Data }</p> <p>通道上报状态::=structure {   通道 OAD,   上报状态 unsigned }</p> <p>上报状态:</p>	<p>用于存储事件记录, 记录根据“触发事件配置参数”自动产生。 记录的顺序按照事件发生的次序排序, 复位后事件记录表为空。 事件记录序号——单调递增。 事件发生源——具体内容由对象实例定义。 事件上报状态——按通道分别记录上报状态。 关联对象属性的数据——其排列次序和个数 n, 由“逻辑名”引用的对象实例的属性 3 决定</p>

表 143（续）

编号	属性	说明
2	bit0: 事件发生上报标识, 0 表示未上报, 1 表示已上报; bit1: 事件发生上报确认标识, 0 表示未确认, 1 表示已确认; bit2: 事件结束 (恢复) 上报标识, 0 表示未上报, 1 表示已上报; bit3: 事件结束 (恢复) 上报确认标识, 0 表示未确认, 1 表示已确认	
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::=OAD OAD 属性特征: 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。调用方法复位后, 记录表中不包含任何记录, 此时当前记录数为零。在每次事件发生执行记录操作后, 当前记录数加 1, 直到记录数等于最大记录数
5	最大记录数::=long-unsigned	
6	配置参数::=structure { 参数 1 instance-specific, ... 参数 n instance-specific }	用于配置触发事件记录的判定参数, 参数的数据类型依据“逻辑名”决定的对象实例而定义, 见 E.4
7	当前值记录表::=array 当前值 当前值::=structure { 事件发生源 instance-specific, 累计时间及发生次数 structure { 事件发生次数 double-long-unsigned, 事件累计时间 double-long-unsigned } }	事件发生源, 在具体的事件对象中定义。 事件发生次数——单位: 次。 事件累计时间——单位: s
8	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) }	
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 144。

表 144 事件对象类方法说明

编号	方法	说明
1	复位（参数） 参数:: $\text{integer}(0)$	通用方法，instance-specific
2	执行（参数） 参数:: $\text{Data}$	通用方法，instance-specific
4	添加一个事件关联对象属性（参数） 参数:: $\text{OAD}$ 对象属性描述符	在属性“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:: $\text{OAD}$ 对象属性描述符	在属性“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

### 7.3.8 参数变量类

参数变量类定义了参数信息，定义见表 145。

表 145 参数变量类定义

参数变量类，class_id=8		$0 \sim n$
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 146。

表 146 参数变量类属性说明

编号	属性	说明
1	逻辑名:: $\text{octet-string}$	
2	参数:: $\text{instance-specific}$	参数的数据类型依据“逻辑名”决定的对象实例而定义

参数变量类方法说明见表 147。

表 147 参数变量类方法说明

编号	方法	说明
1	复位（参数） 参数:: $\text{integer}(0)$	通用方法，instance-specific
2	执行（参数） 参数:: $\text{Data}$	通用方法，instance-specific

## 7.3.9 冻结数据类

冻结数据类定义了配置、存储冻结数据及其信息，定义见表 148。

表 148 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		必选/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 149。

表 149 冻结数据类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
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 }	冻结数据（记录），记录根据“冻结周期”和“延时时间”自动冻结或执行“触发一次冻结”产生。 记录的顺序按照冻结发生的次序排序，复位后冻结数据表为空。 冻结记录序号——单调递增。 关联对象属性的数据——其排列次序和个数 $n$ ，由“逻辑名”引用的对象实例的属性 3 决定
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	具体配置参可由对象实例定义

冻结数据类方法说明见表 150。

表 150 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	根据延时时间（参数）触发执行一次事件记录操作。 参数——延时执行的时间，单位：s。0 表示立即执行（无延时）
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 }	在属性“关联对象属性表”中，增加一个冻结对象属性
5	删除一个冻结对象属性（参数） 参数::=OAD 对象属性描述符	在属性“关联对象属性表”中，删除一个冻结对象属性
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::=NULL	

7.3.10 采集类

采集类定义了配置和存储与采集相关的参数、数据和记录。采集类定义见表 151。

表 151 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 152。

表 152 采集类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	配置表::=array 配置单元 配置参数::=instance-specific	配置单元——配置采集相关的参数，其内容及其数据类型依据“逻辑名”引用的对象实例定义，见 E.7

表 152 (续)

编号	属性	说明
3	记录表::=array 记录单元 记录单元::=instance-specific	是“记录单元”的数组。 记录单元——记录与采集相关的数据，其内容及其数据类型中“逻辑名”引用的对象实例定义，见 E.7

采集类方法说明见表 153。

表 153 采集类方法说明

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	清空记录表 (参数) 参数::=NULL	

### 7.3.11 集合类

集合类对象提供配置、存储终端采集数据及其相关信息，定义见表 154。

表 154 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 155。

表 155 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——其内容及其数据类型由“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，此时当前记录数为零
4	最大元素个数	

集合类方法说明见表 156。

表 156 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

7.3.12 脉冲计量类

脉冲计量类定义见表 157。

表 157 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
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 158。

表 158 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 { TV long-unsigned, TA long-unsigned }	
4	脉冲配置::=array 脉冲单元 脉冲单元::=structure { 脉冲输入端口号 OAD, 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) }, 脉冲常数 k long-unsigned }	
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	包含总及 $n$ 个费率的电能量, 规定其第一个数组元素是总电能量, 后面依次排列费率 $1 \cdots n$ 电能量
8	当月正向有功电量::=array 电能量 电能量::=double-long-unsigned	
9	当日反向有功电量::=array 电能量 电能量::=double-long-unsigned	
10	当月反向有功电量::=array 电能量 电能量::=double-long-unsigned	
11	当日正向无功电量::=array 电能量 电能量::=double-long-unsigned	
12	当月正向无功电量::=array 电能量 电能量::=double-long-unsigned	
13	当日反向无功电量::=array 电能量 电能量::=double-long-unsigned	
14	当月反向无功电量::=array 电能量 电能量::=double-long-unsigned	
15	正向有功电能示值::=array 电能示值 电能示值::=double-long-unsigned	

表 158（续）

编号	属性	说明
16	反向有功电能示值::=array 电能示值 电能示值::=double-long-unsigned	
17	正向无功电能示值::=array 电能示值 电能示值::=double-long-unsigned	
18	反向无功电能示值::=array 电能示值 电能示值::=double-long-unsigned	
19	单位及换算::=structure { 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）， 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） }	Scaler_Unit 见 6.3.3.14

脉冲计量类的方法描述见表 159。

表 159 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（脉冲输入端口号）	

7.3.13 控制类

控制类定义了用电负荷控制的功能，定义见表 160。

表 160 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

表 160 (续)

控制类, class_id=13		0~n
属性		数据类型
4. 控制输出状态	(dyn.)	array
5. 超限告警状态	(dyn.)	array
方法		必选/可选
1. 复位		可选
2. 执行		可选
3. 添加控制单元		必选
4. 删除控制单元		必选
5. 更新控制单元		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 161。

表 161 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 控制单元::=instance-specific	控制方案单元——其内容及其数据类型依据“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 1~8 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	超限告警状态::=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 162。



表 162 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	添加控制单元（控制方案单元） 控制方案单元::=instance-specific	
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	

### 7.3.14 区间统计类

区间统计类定义了统计越限相关的信息，定义见表 163。

表 163 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 164。

表 164 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 {	“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 s

表 164（续）

编号	属性	说明
2	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 }	
3	关联对象属性表::=array 一个关联对象属性 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， 统计频率 TI }	统计频率：采样点取值周期

区间统计类方法说明见表 165。

表 165 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	添加一个统计对象属性（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，增加一个统计对象属性
4	删除一个统计对象属性（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

7.3.15 累加平均类

累加平均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 166。

表 166 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

表 166（续）

累加平均类，class_id=15		0~n
属性		数据类型
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. 添加一个关联对象属性		可选
4. 删除一个关联对象属性		可选

累加平均类属性说明见表 167。

表 167 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——其数据类型依据“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, 统计频率 TI }	

累加平均类方法说明见表 168。

表 168 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	添加一个关联对象属性（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

表 168（续）

编号	方法	说明
4	删除一个关联对象属性（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

7.3.16 极值工具类

极值工具类对象提供采集或生成最大、最小值及其发生时间的功能，定义见表 169。

表 169 极值工具类定义

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

极值工具类属性说明见表 170。

表 170 极值工具类属性说明

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, 及其发生时间 date_time_s, 最小值 instance-specific, 及其发生时间 date_time_s }	最大值、最小值——其数据类型依据“逻辑名”引用的对象实例而定义
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, 统计频率 TI }	

极值工具类方法说明见表 171。

表 171 极值工具类方法说明

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	添加一个关联对象属性（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	删除一个关联对象属性（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

7.3.17 显示类

显示类定义了与显示相关的信息，定义见表 172。

表 172 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 173。

表 173 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，其他为分屏显示（1 表示分屏第一屏，2 表示分屏第二屏，以此类推）

表 173 (续)

编号	属性	说明
3	每个对象显示时间: long-unsigned	显示时长, 单位为 s, 0 表示由外部触发
4	显示参数::=structure { 当前总对象数      unsigned, 允许最大对象数   unsigned }	当前总对象数——指示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

显示类方法说明见表 174。

表 174 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: 单位为 s
6	全显 (参数) 参数::=long-unsigned	参数为全显持续时间, 单位为 s

### 7.3.18 文件传输类

文件传输类定义了服务器实现上传和下载文件的功能, 定义见表 175。

表 175 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. 命令结果	(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

表 175（续）

方法	必选/可选
5. 代发	可选
6. 代收	可选
7. 上传	可选
8. 下载	可选

文件传输类属性说明见表 176。

表 176 文件传输类属性说明

编号	属性	说明
1	逻辑名	标记接口类对象实例，使用 OI，详见 7.4
2	<p>文件信息::=structure</p> <pre>{   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)   } }</pre> <p>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）</p>	<p>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：单位为 B</p>
3	<p>命令结果::=enum</p> <pre>{   文件传输进度 0~99%                (0~99),   传输或执行操作成功                (100),   正在建立连接（扩展传输）          (101),   正在远程登录（扩展传输）          (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),   建立连接失败（扩展传输）          (110),   远程登录失败（扩展传输）          (111),   存储空间不足                      (112),   复位后默认值                      (255) }</pre>	当前文件传输的过程状态

文件传输类方法说明见表 177。

表 177 文件传输类方法说明

编号	方法	说明
1	复位（参数） 参数::=integer (0)	强迫对象复位。将属性“数值”置为默认值，默认值是实例的常数
2	执行（参数） 参数::=Data	通用方法，instance-specific。 默认方法执行下载文件，如果文件有执行权限则执行，否则拒绝
3	删除（参数） 参数::=NULL	删除本地文件，如果文件存在则删除，并复位传输状态字和当前文件指针，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 A
5	代发（参数） 参数::=TSA { 代发至其他服务器 [0] TSA, 代发至端口 [1] OAD }	文件下载到本地端后，再根据目标地址进行文件代发

## 7.3.19 设备管理类

设备管理类定义见表 178。

表 178 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
7. 允许跟随上报	(static)	bool



表 178 (续)

设备管理类, class_id=19		0~n
属性		数据类型
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
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

设备管理类属性说明见表 179。

表 179 设备管理类属性说明

编号	属性	说明
1	逻辑名	
2	设备描述符::=visible-string	
3	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)) , 厂家扩展信息   visible-string (SIZE (8)) }	
4	生产日期::=date_time_s	
5	子设备列表::=array 0I	
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	

设备管理类方法说明见表 180。

表 180 设备管理类方法说明

编号	方法	说明
1	复位（参数） 参数:: <code>=NULL</code>	设备复位重启，参数 NULL
2	执行（参数） 参数:: <code>=Data</code>	通用方法，instance-specific
3	数据初始化	清空设备数据区
4	恢复出厂参数（参数） 参数:: <code>=array OAD</code>	将设备的配置恢复到出厂设置。 参数定义的 OAD 列表保持此状，不在恢复之列保持不变
5	事件初始化（参数） 参数:: <code>=NULL</code>	清空所有事件存储区
6	需量初始化（参数） 参数:: <code>=NULL</code>	当前需量对象清零

## 7.3.20 应用连接类

应用连接类定义见表 181。

表 181 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 182。

表 182 应用连接类属性说明

编号	属性	说明
1	逻辑名:: <code>=octet-string</code>	
2	对象列表:: <code>=array</code> 一个可访问对象 一个可访问对象:: <code>=structure</code> { 对象标识 OI, 访问权限 structure } 访问权限:: <code>=structure</code> { 属性访问权限 array 一个属性访问权限，	包含对象所有可访问对象以及该对象属性和方法的访问权限

表 182（续）

编号	属性	说明
2	<pre>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 {     方法 ID          unsigned,     方法访问权限      bool }</pre>	
3	<pre>应用语境信息::=structure {     协议版本信息      long-unsigned,     最大接收 APDU 尺寸 long-unsigned,     最大发送 APDU 尺寸 long-unsigned,     最大可处理 APDU 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 }</pre>	
4	当前连接的客户机地址::=unsigned	
5	<pre>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) }</pre>	见 6.3.6.2

应用连接接口类方法说明见表 183。

表 183 应用连接接口类方法说明

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

7.3.21 ESAM 接口类

ESAM 接口类定义参见表 184。

表 184 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
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
方法		必选/可选
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 185。

表 185 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

表 185 (续)

编号	属性	说明
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: min
6	会话时效剩余时间::=double-long-unsigned	单位: min
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 { 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 4B 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: min
15	红外认证剩余时间::=double-long-unsigned	单位: min

ESAM 接口类方法说明见表 186。

表 186 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 操作 (参数) 参数::=SID 应答::=octet-string	对 ESAM 数据的操作, 电能表透传给 ESAM
4	数据更新 (参数) 参数::=structure { 参数内容 octet-string, 数据验证码 SID_MAC } 参数内容格式定义: OAD (4B) + LEN (1B) + 内容 (见 ESAM 文件结构)	数据、数据 MAC 先发给 ESAM 验证, 验证成功后, 再设置到终端、电能表中

表 186 (续)

编号	方法	说明
5	协商失效 (参数) 参数::=NULL 应答::=date_time_s	应答时间内当前日期时间
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, 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	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
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	

7.3.22 输入输出设备类

输入输出设备类定义了输入输出设备相关信息，定义见表 187。

表 187 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
方法		必选/可选
1. 复位		可选
2. 执行		可选

输入输出设备类的属性描述见表 188。

表 188 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

输入输出设备类的方法描述见表 189。

表 189 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

7.3.23 总加组类

总加组类定义了总加组相关信息，定义见表 190。

表 190 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. 总加配置表	(static)	array
3. 总加有功功率	(dyn.)	long64

表 190 (续)

属性		数据类型
4. 总加无功功率	(dyn.)	long64
5. 总加滑差时间内平均有功功率	(dyn.)	long64
6. 总加滑差时间内平均无功功率	(dyn.)	long64
7. 总加日有功电量	(dyn.)	array
8. 总加日无功电量	(dyn.)	array
9. 总加月有功电量	(dyn.)	array
10. 总加月无功电量	(dyn.)	array
11. 总加剩余电量 (费)	(dyn.)	long64
12. 当前功率下浮控控后总加有功功率冻结值	(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. 清空总加配置单元		必选
2. 执行		可选
3. 添加一个总加配置单元		可选
4. 批量添加总加配置单元		可选
5. 删除一个总加配置单元		可选

总加组类的属性描述见表 191。

表 191 总加组类属性说明

编号	属性	说明
1	逻辑名::=octet-string	
2	总加配置表::=array 总加组配置单元 总加配置单元::=structure { 参与总加的分路通信地址 TSA, 总加标识 enum{正向(0), 反向(1)}, 运算符标识 enum{加(0), 减(1)} }	
3	总加有功功率::=long64	
4	总加无功功率::=long64	
5	总加滑差时间内平均有功功率::=long64	
6	总加滑差时间内平均无功功率::=long64	



表 191 (续)

编号	属性	说明
7	总加日有功电量::=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, 后面依次排列费率 1~ $n$ 电能量
8	总加日无功电量::=array 电能量 电能量::=long64	
9	总加月有功电量::=array 电能量 电能量::=long64	
10	总加月无功电量::=array 电能量 电能量::=long64	
11	总加剩余电量(费)::=long64	
12	当前功率下浮控控后总加有功功率冻结值::=long64	
13	总加组滑差时间周期::=unsigned	单位: min
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)) 功控轮次标识位: 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, 功控时段有效标识位       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 } 功控时段有效标识位: 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)): 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	时段控定值方案号: 表示所投入的功控定值方案号

表 191 (续)

编号	属性	说明
17	<p>总加组当前控制状态::=structure</p> <pre> {     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 } </pre> <p>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” :  未处于相应状态。  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”未处于相应状态。  bit0: 月电控;  bit1: 购电控;  bit2~bit7: 备用</p>	
18	<p>换算及单位::=structure</p> <pre> {     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) ,     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) } </pre>	Scaler_Unit 见 6.3.3.14

总加组类的方法描述见表 192。

表 192 总加组类方法说明

编号	方法	说明
1	<p>清空总加配置表 (参数)  参数::=NULL</p>	清空总加配置表
2	<p>执行 (参数)  参数::=Data</p>	通用方法, instance-specific
3	<p>添加一个总加配置单元 (参数)  参数::=总加配置单元</p>	向总加配置表中添加一个总加配置单元

表 192 (续)

编号	方法	说明
4	批量添加总加配置单元 (参数) 参数::=array 总加配置单元	向总加配置表中添加若干个总加配置单元
5	删除一个总加配置单元 (参数) 参数::=参与总加的分路通信地址 TSA	删除总加配置表中的一个总加配置单元

### 7.3.24 分项事件对象类

分项事件类定义了配置、存储分项事件数据信息，定义见表 193。

表 193 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
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 194。

表 194 分项事件对象类属性说明

编号	属性	说明
1	逻辑名::=octet-string	

表 194（续）

编号	属性	说明
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 128
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。执行方法“复位”后，记录表清空，此时当前记录数为零
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 E.4
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 143。 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 { 事件发生次数 double-long-unsigned, 事件累计时间 double-long-unsigned }	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：s 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	

表 194 (续)

编号	属性	说明
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 195。

表 195 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) 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
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

### 7.3.25 无线公网通信接口类

无线公网通信接口类定义了无线公网通信接口, 定义见表 196。

表 196 无线公网通信接口类定义

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
5. 版本信息	(static)	structure
6. 支持规约列表	(static)	array
7. SIM 卡的 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
方法		必选/可选
1. 复位		可选

无线公网通信接口类属性说明见表 197。

表 197 无线公网通信接口类属性说明

编号	属性	说明
1	逻辑名	
2	<pre> 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     (         bit7~bit2: 超时时间 (s),         bit1~bit0: 重发次数     ),     心跳周期 (秒)  long-unsigned } </pre>	
3	<pre> 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, } </pre>	
4	<pre> 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)) } </pre>	
5	<pre> 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)) ,     厂家扩展信息    visible-string (SIZE (8)) } </pre>	
6	支持规约列表::=array visible-string	
7	SIM 卡的 ICCID::=visible-string (SIZE (20))	
8	IMSI::=visible-string (SIZE (15))	

表 197（续）

编号	属性	说明
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	

无线公网通信接口类方法说明见表 198。

表 198 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

7.3.26 以太网通信接口类

以太网通信接口类定义了以太网通信接口，定义见表 199。

表 199 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 200。

表 200 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），} }	连接应用方式： 主备模式：按主站通信参数表次序依次连接 主站，只连接一个主站； 多连接模式：按照主站通信参数同时连接多 个主站

表 200（续）

编号	属性	说明
2	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 ( bit7~bit2: 超时时间 (s) , bit1~bit0: 重发次数 ) , 心跳周期 (s) 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 201。

表 201 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

7.4 对象标识

7.4.1 对象标识格式定义

对象标识 (OI) 由 2B 组成，采用十六进制编码表示，其格式定义见图 26。

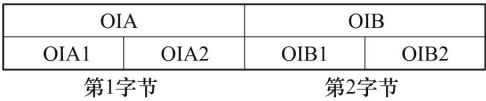


图 26 对象标识 (OI) 格式定义



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。对象标识定义见附录 E。

#### 7.4.2 电能量类对象标识

电能量类对象的标识定义见表 202。

表 202 电能量类对象标识定义

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 AH: 反向视在	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	

#### 7.4.3 最大需量类对象标识

最大需量类对象的标识定义见表 203。

表 203 最大需量类对象标识定义

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 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	0H: 合相 1H: A 相 2H: B 相 3H: C 相

#### 7.4.4 变量类对象标识

变量类对象的标识定义见表 204。

表 204 变量类对象标识定义

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: 1min 平均有功功率 08H: 1min 平均无功功率 09H: 1min 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: 电压谐波含有量 (总及 2~n 次) 0EH: 电流谐波含有量 (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 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: 负荷率 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	
		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	

表 204 (续)

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
2H: 变量	1H: 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 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	

## 7.4.5 事件类对象标识

事件类对象的标识定义见表 205。

表 205 事件类对象标识定义

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: 过负荷 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 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	
	1H: 采集	00H: 终端初始化 01H: 终端版本变更	

表 205 (续)

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: 输出回路开关接入状态变位记录 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: 终端对电表校时记录 1CH: 电能表数据变更监控记录	
	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: 电报告警事件记录	
	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 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: 有功总电能量差动超限事件记录单元 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 20H: 新增上报事件列表	

## 7.4.6 参变量类对象标识

参变量类对象的标识定义见表 206。

表 206 参变量类对象标识定义

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: 备用套日时段表切换时间 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: 计量原件数 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 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: 其他金额限值 20H: 报警电量限值 21H: 其他电量限值 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 30H: 电压合格率参数	
	1: 计量	00H: 最大需量周期 01H: 滑差时间 02H: 校表脉冲宽度 03H: 资产管理编码 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	

表 206 (续)

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
4H: 参变量	1: 计量	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 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	
	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	
		20H: 公网远程通信多接入点备用通道	

#### 7.4.7 冻结类对象标识

冻结类对象的标识定义见表 207。

表 207 冻结类对象标识定义

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	

表 207 (续)

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
5H: 冻结	0H: 通用	03H: 小时冻结 04H: 日冻结 05H: 结算日冻结 06H: 月冻结 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	

## 7.4.8 采集监控类对象标识

采集监控类对象的标识定义见表 208。

表 208 采集监控类对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
6H: 采集监控	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 12H: 任务配置表 13H: 任务配置单元 14H: 普通采集方案集合 15H: 普通采集方案 16H: 事件采集方案集合 17H: 事件采集方案 18H: 透明方案集合 19H: 透明方案 1AH: 透明方案结果集 1BH: 一个透明方案结果 1CH: 上报方案集合 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	

## 7.4.9 集合类对象标识

集合类对象的标识定义见表 209。



表 209 集合类对象标识定义

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	

## 7.4.10 控制类对象标识

控制类对象的标识定义见表 210。

表 210 控制类对象标识及对应接口类定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
8H: 控制	0H: 通用	00H: 遥控 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 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	

## 7.4.11 文件传输类对象标识

文件传输类对象的标识定义见表 211。

表 211 文件传输类对象标识定义

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	

## 7.4.12 ESAM接口类对象标识

ESAM 接口类对象的标识定义见表 212。

表 212 ESAM 接口类对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
FH: 其他	1H: 安全	00H: ESAM	
		01H: 安全模式参数	

## 7.4.13 输入输出设备接口类对象标识

输入输出设备接口类对象的标识定义见表 213。

表 213 输入输出设备接口类对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
FH: 其他	2H: 输入输出设备	00H: RS-232 01H: RS-485 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: 交流采集接口 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 10H: 从节点单元 11H: 网络拓扑单元	

## 7.4.14 显示类对象标识

显示类对象的标识定义见表 214。

表 214 显示类对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
FH: 其他	3H: 显示	00H: 自动轮显 01H: 按键轮显	

## 7.4.15 厂家自定义对象

显示类对象的标识定义见表 215。

表 215 显示类对象标识定义

对象标识 (OI)			
OIA1	OIA2	OIB1	OIB2
FH: 其他	FH: 厂家自定义		

# 附录 A

## (资料性附录)

### 校验算法

#### A.1 校验算法

```

/*
 * u16 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 u16;
/*
 * FCS lookup table as calculated by the table generator.
 */
static u16 fctab [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,
    0xbdc b, 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,
    0xdec d, 0xcf44, 0xfddf, 0xec56, 0x98e9, 0x8960, 0xbbfb, 0xaa72,
    0x6306, 0x728f, 0x4014, 0x519d, 0x2522, 0x34ab, 0x0630, 0x17b9,
    0xef4e, 0xfec7, 0xcc5c, 0xdd d5, 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) ^ fctab [(fcs ^ *cp++) & 0xff];
    return (fcs);
}
/*
 * How to use the fcs
 */
tryfcs16 (cp, len)
register unsigned char *cp;
register int len;
{
    u16 trial_fcs;
    /* add on output */
    trial_fcs = pppfcs16 ( PPPINITFCS16, cp, len );
    trial_fcs ^= 0xffff; /* complement */
    cp [len] = (trial_fcs & 0x00ff); /* least significant byte first */
    cp [len+1] = ((trial_fcs >> 8) & 0x00ff);
    /* check on input */
    trial_fcs = pppfcs16 ( PPPINITFCS16, cp, len + 2 );
    if ( trial_fcs == PPPGOODFCS16 )
        printf ("Good FCS\n");
}

```

## A.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:   $x^{**0} + x^{**5} + x^{**12} + x^{**16}$ .
 */
#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
 * ( $x^{**0} + x^{**5} + x^{**12} + x^{**16}$ )
 */
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");
}

```

附 录 B  
(规范性附录)  
物 理 单 位 枚 举

物理单位的枚举见表 B.1。

表 B.1 物理单位的枚举

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	$7 \times 24 \times 60 \times 60\text{s}$
4	d	时间	天	$24 \times 60 \times 60\text{s}$
5	h	时间	小时	$60 \times 60\text{s}$
6	min	时间	分	$60\text{s}$
7	s	时间 ( $t$ )	秒	s
8	(°)	角度	度	$\text{rad} \times 180/\pi$
9	°C	温度 ( $T$ )	摄氏度	$\text{K}-273.15$
10	CNY	(当地) 货币		
11	m	长度 ( $l$ )	米	m
12	m/s	速度 ( $v$ )	米每秒	m/s
13	m <sup>3</sup>	体积 ( $V$ ) $r_V$ , 仪表常数或脉冲值 (容积)	立方米	m <sup>3</sup>
14	m <sup>3</sup>	修正的体积	立方米	m <sup>3</sup>
15	m <sup>3</sup> /h	流量	立方米每小时	m <sup>3</sup> / (60×60s)
16	m <sup>3</sup> /h	修正的流量	立方米每小时	m <sup>3</sup> / (60×60s)
17	m <sup>3</sup> /d	流量	立方米每天	m <sup>3</sup> / (24×60×60s)
18	m <sup>3</sup> /d	修正的流量	立方米每天	m <sup>3</sup> / (24×60×60s)
19	L	容积	升	$10^{-3}\text{m}^3$
20	kg	质量 ( $m$ )	千克	kg
21	N	力 ( $F$ )	牛顿	N
22	Nm	能量	牛顿米	$\text{J}=\text{Nm}=\text{Ws}$
23	Pa	压强 ( $p$ )	帕斯卡	$\text{N}/\text{m}^2$
24	bar	压强 ( $p$ )	巴	$10^{-5}\text{N}/\text{m}^2$
25	J	能量	焦耳	$\text{J}=\text{Nm}=\text{Ws}$
26	J/h	热功	焦每小时	$\text{J}/(60 \times 60\text{s})$
27	W	有功功率 ( $P$ )	瓦	$\text{W}=\text{J}/\text{s}$
28	kW	有功功率 ( $P$ )	千瓦	$\text{kW}=\text{J}/(\text{s} \times 1000)$
29	VA	视在功率 ( $S$ )	伏安	
30	kVA	视在功率 ( $S$ )	千伏安	

表 B.1 (续)

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	$\text{kW} \times (60 \times 60\text{s})$
34	kVAh	视在能量 $r_s$ , 视在电能表常数或脉冲值	千伏安时	$\text{kVA} \times (60 \times 60\text{s})$
35	kvarh	无功能量 $r_B$ , 无功电能表常数或脉冲	千乏时	$\text{kvar} \times (60 \times 60\text{s})$
36	A	电流 $I$	安	A
37	C	电量 $Q$	库仑	$\text{C} = \text{As}$
38	V	电压 $U$	伏	V
39	V/m	电场强度 $E$	伏每米	V/m
40	F	电容 $C$	法	$\text{C/V} = \text{As/V}$
41	$\Omega$	电阻 $R$	欧姆	$\Omega = \text{V/A}$
42	$\Omega\text{m}$	电阻系数 $\rho$	欧姆米	$\Omega\text{m}$
43	Wb	磁通量 $\Phi$	韦伯	$\text{Wb} = \text{Vs}$
44	T	磁通密度 $B$	特斯拉	$\text{Wb/m}^2$
45	A/m	磁场强度 $H$	安每米	A/m
46	H	电感 $L$	亨利	$\text{H} = \text{Wb/A}$
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	B	存储容量	字节	
53	dBmW	发射功率	分贝毫瓦	
54	元/kWh	电价	元每千瓦时	
55	Ah	电池容量	安时	
56	ms	时间	毫秒	
57	dB $\mu$ V	发射电平	分贝微伏	
58~253	保留			
254	其他单位			
255	无单位、缺单位、计数			

数值举例见表 B.2。



表 B.2 数 值 举 例

数值	换算	单位	数据
263783	—4	kWh	26.3783kWh
60201	—3	A	60.201A
22000	—2	V	220V
1	—1	℃	0.1℃
3467	0	V	3467V
593	3	Wh	593kWh

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 C.1。

表 C.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),     DL/T 645—2007                                (30),     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 C.2。

表 C.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), </pre>	

表 C.2 (续)

数据类型定义		说明
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) ,	
级联	(24) ,	
直流模拟量	(25) ,	
弱电端子 12V 输出	(26) ,	
搜表	(27) ,	
三相负荷平衡	(28) ,	
升级	(29) ,	
}		

## 附录 D (资料性附录) APDU 编码举例

### D.1 预连接

#### D.1.1 登录

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

68 —— 帧起始符

1E 00 —— 长度域 L=30B

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=48B

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

#### D.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。

## D.2 建立应用连接

发送: 02 00 00 10 FF 04

00 04 00 01 04 00 00 00 00 64 00 00

02 —— [2] CONNECT-Request

00 —— PIID

00 10 —— 期望的应用层协议版本号

FF FF FF FF FF FF FF FF —— ProtocolConformance

FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 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 04 00 04 00 01

04 00 00 00 00 64 00 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)) + 厂家扩展信息 (size (8))

00 10 —— 期望的应用层协议版本号

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

### D.3 读取

#### D.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

#### D.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 OAD, 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

### D.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

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 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 06 00 00 00 00 00 00

85 —— [133] GET-Response  
 03 —— [3] GetResponseRecord  
 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



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 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  
 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 00 00 00 00 00 00

85 —— [133] GET-Response

03 —— [3] GetResponseRecord

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  
 0102

010506000000000060000000000600000000060000000000600000000

010506000000000060000000000600000000060000000000600000000

09050000000122 1C07E00114000000 1C07E00114000000 1C07E00114000000  
 0102

010506000000000060000000000600000000060000000000600000000

010506000000000060000000000600000000060000000000600000000

09050000000123 1C07E00114000000 1C07E00114000000 1C07E00114000000  
 0102

010506000000000060000000000600000000060000000000600000000

010506000000000060000000000600000000060000000000600000000

09050000000124 1C07E00114000000 1C07E00114000000 1C07E00114000000  
 0102

010506000000000060000000000600000000060000000000600000000

010506000000000060000000000600000000060000000000600000000

09050000000125 1C07E00114000000 1C07E00114000000 1C07E00114000000

0102

0105060000000006000000000600000000060000000006000000000

0105060000000006000000000600000000060000000006000000000

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

00 —— 没有时间标签

### D.3.4 读取多个记录型对象属性

参考读取一个记录型对象属性。

### D.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 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 12 00 01 12 00 01 01 00 00 00

85 —— [133] GET-Response

05 —— [5] GetResponseNext

08 —— PIID-ACD

00 —— 末帧标识 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 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 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 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 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 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 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 00 09 06 00 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 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

09 —— PIID-ACD

01 —— 末帧标识 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 —— array

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 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 00 09 06 00 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 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

### D.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:

发送帧: 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

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

响应帧: 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  
 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

## D.4 设置

### D.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

#### D.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 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

#### D.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 3s  
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

## D.5 操作

### D.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  
 05 —— PIID-ACD  
 00 10 01 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

### D.5.2 操作多个对象方法请求

参考操作一个对象方法。

### D.5.3 操作多个对象方法后读取多个对象属性请求

执行电能量复位方法后读取当前电能量总及分费率  
 发送: 07 03 07 01 00 10 01 00 0F 00 00 10 02 00 00 00  
 07 —— [7] ACTION-Request  
 03 —— [3] ActionThenGetRequestNormalList  
 07 —— PIID  
 01 —— SEQUENCE OF 的个数=1

00 10 01 00 —— OMD

0F 00 —— 参数 Data, integer (0)

00 10 02 00 —— OAD

00 —— 读取延时, 0 表示服务器默认的延时

00 —— 没有时间标签

响应: 87 03 07 01 00 10 01 00 00 00 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 00

87 —— [135] ACTION-Response

03 —— [3] ActionThenGetResponseNormalList

07 —— PIID-ACD

01 —— SEQUENCE OF 的个数=1

00 10 01 00 —— OMD

00 —— DAR, 0 成功

00 —— Data OPTIONAL=0 表示没有数据

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

#### D.5.4 添加普通采集方案

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 —— 0AD 00100200

5B 00 00 20 02 00 —— 0AD 00200200

5B 00 20 00 02 00 —— 0AD 20000200

5B 00 20 01 02 00 —— 0AD 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

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 —— 电能表集合数据类型  
01 —— 电能表集合：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

## D.6 上报

### D.6.1 上报多个记录型对象属性

参考读取多个记录型对象属性。

## D.7 代理

### D.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表

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 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

#### D.7.2 代理读取一个服务器的一个记录型对象属性

参考读取一个记录型对象属性。

#### D.7.3 代理设置多个服务器的多个对象属性

参考设置多个对象属性。

**D.7.4 代理设置后读取多个服务器的多个对象属性**

参考设置后读取多个对象属性。

**D.7.5 代理操作多个服务器的多个对象方法**

参考操作多个对象方法。

**D.7.6 代理操作后读取多个服务器的多个对象方法和属性**

参考操作多个对象方法后读取多个对象属性。

附 录 E  
(规范性附录)  
对 象 标 识 定 义

### E.1 电能量类对象

有关电能量类的对象标识定义见表 E.1。

表 E.1 电能量类对象标识定义

对象标识 OI	接口类 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

表 E.1 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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
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
00A0	1	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

表 E.1 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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
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
0223	1	C 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

表 E.1 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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
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

## E.2 最大需量类对象

有关最大需量类的对象标识定义见表 E.2。

表 E.2 最大需量类对象标识定义

对象标识 OI	接口类 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

表 E.2 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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; 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



表 E.2 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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
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

表 E.2 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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; 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

表 E.2 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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
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。			

E.3 变量类对象

有关变量类的对象标识定义见表 E.3。

表 E.3 变量类对象标识定义

对象标识 OI	接口类 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
2002	3	电压相角	数据类型: long-unsigned; 单位: (°); 换算: -1
2003	3	电压电流相角	数据类型: long-unsigned; 单位: (°); 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	1min 平均有功功率	数据类型: double-long; 单位: W; 换算: -1
2008	4	1min 平均无功功率	数据类型: double-long; 单位: var; 换算: -1
2009	4	1min 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

表 E.3 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
200C	3	电流波形失真度	数据类型: long; 单位: %; 换算: -2
200D	5	电压谐波含有量 (总及 2~n 次)	数据类型: long; 单位: %; 换算: -2
200E	5	电流谐波含有量 (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	数据类型: 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: min; 无换算
2014	6	电能表运行状态字	数据类型: array bit-string; 无单位, 无换算, 包括电能表运行状态字 1~7, 见附录 F
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 F 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))
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: s; 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	负荷率	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 相安时值包总、A、B、C 相
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 }

表 E.3 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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
2100	14	分钟区间统计	统计周期, 单位为分钟 (min)
2101	14	小时区间统计	统计周期, 单位为小时 (h)
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	分钟平均	统计周期, 单位为分钟 (min)
2111	15	小时平均	统计周期, 单位为时
2112	15	日平均	统计周期, 单位为日
2113	15	月平均	统计周期, 单位为月
2114	15	年平均	统计周期, 单位为年
2120	16	分钟极值	统计周期, 单位为分钟 (min)
2121	16	小时极值	统计周期, 单位为时
2122	16	日极值	统计周期, 单位为日
2123	16	月极值	统计周期, 单位为月
2124	16	年极值	统计周期, 单位为年
2131	6	当月 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, 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: min; 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: min; 无换算), 电压超下限时间   double-long-unsigned (单位: min; 无换算) }
2132	6	当月 B 相电压合格率	
2133	6	当月 C 相电压合格率	
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	

表 E.3 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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: min; 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 }
2301	23	总加组 1	
2302	23	总加组 2	
2303	23	总加组 3	
2304	23	总加组 4	
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: m <sup>3</sup> ; 换算: -4 (水表和热表共用)
2501	6	累计气流量	数据类型, double-long-unsigned; 单位: m <sup>3</sup> ; 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: h; 换算: 0

表 E.3 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
2505	6	水温	数值::=structure { 供水温度  double-long-unsigned, 回水温度  double-long-unsigned } 单位: °C; 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)} }
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。 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值, 正值代表输入, 负值代表输出。表内温度正值为零上(摄氏度), 负值为零下(摄氏度)。 注 3: 相角测量范围是 $0^\circ \sim 360^\circ$ 。 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平均功率。 注 5: 组合无功最大需量的正负值是用来标识潮流的方向, 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, 如果来自象限三、四, 以负值表示。			

#### E.4 事件类对象

有关事件类的对象标识定义见表 E.4。

表 E.4 事件类对象标识定义

对象标识 OI	接口类 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: s; 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 }
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) }

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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), 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 { 电压触发限值 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	电能表过负荷事件	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 { 触发下限 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



表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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 { 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) } 事件发生源::=NULL

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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	电能表周休日编程事件	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 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

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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 { } 事件发生源 ::= NULL
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 ::= NULL
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记录单元 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) }

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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 { } 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
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

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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 { } 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
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 { } 事件发生源 ::= NULL
3101	7	终端版本变更事件	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
3104	7	终端状态量变位事件	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
3105	7	电能表时钟超差事件	属性 2 (事件记录表) ::= array 电能表时钟超差记录单元 属性 6 (配置参数) ::= structure { 异常判别阈值 long-unsigned (单位: s) , 关联采集任务号 unsigned } 事件发生源 ::= TSA 采集监控任务中需要配置相关 OAD 的采集任务
3106	7	终端停/上电事件	属性 2 (事件记录表) ::= array 停/上电事件记录单元 属性 6 (配置参数) ::= structure { 停电数据采集配置参数 structure { 采集标识 bit-string (SIZE (8)) , 停电事件抄读时间间隔 (h) unsigned , 停电事件抄读时间限值 (min) unsigned, 需要读取停电事件电能表 array TSA }, 停电事件甄别限值参数 structure { 停电时间最小有效间隔 (min) long-unsigned, 停电时间最大有效间隔 (min) long-unsigned, } }

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
3106	7	终端停/上电事件	停电事件起止时间偏差限值 (min) long-unsigned, 停电事件时间区段偏差限值 (min) long-unsigned, 停电发生电压限值 long-unsigned (单位: V; 换算: -1), 停电恢复电压限值 long-unsigned (单位: V; 换算: -1) } } 采集标识: bit0: 置“1”有效, 置“0”无效; bit1: 置“1”随机选择测量点, 置“0”只采集设置的测量点。 事件发生源::=NULL
3107	7	终端直流模拟量越上限事件	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 { 直流模拟量上限 double-long } 事件发生源::=OAD (直流模拟量号)
3108	7	终端直流模拟量越下限事件	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 { 直流模拟量下限 double-long } 事件发生源::=OAD (直流模拟量号)
3109	7	终端消息认证错误事件	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
310A	7	设备故障记录	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::=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) } }
310B	7	电能表示数下降事件	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

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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
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: 字节) } 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

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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, 参与差动的电能的时间区间及对比方法标识 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) } 参与差动的电能的时间区间及对比方法标识: bit0~bit1 编码表示电能的时间跨度, 取值范围 0~2 依次表示 60min 电量、30min 电量、15min 电量, 其他值无效。 bit7 表示对比方法标识: 置“0”, 相对对比, 公式见式 (1); 置“1”, 绝对对比, 公式见式 (2)。 bit2~bit6 备用。 $\frac{ Q-q }{q} \times 100\% \quad (1)$ $ Q-q  \quad (2)$ 式中: $Q$ ——对比的总加组总电能; $q$ ——参照的总加组总电能
3117	7	输出回路接入状态变位事件记录	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 { }
3118	7	终端编程记录	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 { }
3119	7	终端电流回路异常事件	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)}
311A	7	电能表在网状态切换事件	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 此事件只记录电表在网状态变迁



表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
311B	7	终端对电能表校时记录	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 {}
311C	7	电能表数据变更监控记录	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
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	电控告警事件记录	属性 2 (事件记录表) ::= array 电控告警事件单元 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 143
3301	8	标准事件记录单元	标准事件记录单元 ::= 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      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 }
3302	8	编程记录事件单元	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,

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
3302	8	编程记录事件单元	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 }
3303	8	发现未知电能表事件单元	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 }
3304	8	跨台区电能表事件单元	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, ... 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 2min 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, }

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
3306	8	电控跳闸记录单元	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, 跳闸时总加电能量为总加月电能量; 当事件发生源类型为购电控 8107, 跳闸时总加电能量为剩余电量/费
3307	8	电报告警事件单元	电报告警事件单元::=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, 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, 属性标识      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 } 属性标识:

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
3309	8	停/上电事件记录单元	bit0 置“1”为事件正常，bit0 置“0”为事件异常，此处是对停电时间有效间隔的判断； bit1 置“1”为事件有效，bit1 置“0”为事件无效，此处是对停电时间偏差限值的判断； 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, 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 2min 总加组功率 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: s; 无换算) }

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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 { 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”
3310	8	异常插卡事件记录单元	异常插卡事件记录单元::=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, 卡序列号           octet-string, 插卡错误信息字     unsigned, 插卡操作命令头     octet-string, 插卡错误响应状态   long-unsigned, 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 }
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, ...

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
3311	8	退费事件记录单元	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, 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: 模块对应的通信地址仅对表端模块或采集器端模块有效
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, 事件发生时间            date_time_s, ...

表 E.4 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
3315	8	电能表节假日编程 事件记录单元	事件结束时间 <code>date_time_s</code> , 事件发生源 <code>NULL</code> , 事件上报状态 <code>array</code> 通道上报状态, 编程节假日对象 <code>OAD</code> , 编程前节假日内容 公共假日, 第 1 个关联对象属性的数据 <code>Data</code> , ... 第 $n$ 个关联对象属性的数据 <code>Data</code> } 公共假日::= <code>structure</code> { 日期 <code>date</code> , 日时段表号 <code>unsigned</code> }
3320	8	新增上报事件列表	属性 2 (新增上报事件列表, 只读) ::= <code>array OAD</code> 属性 3 (需上报事件对象列表, 只读) ::= <code>array OI</code>  注: 新增上报事件列表对象按通信通道 (OAD) 记录, 当该列表中的事件记录通过“当前”通道被读取后, 从该列表中删除此对象

### E.5 参变量类对象

有关参变量类的对象标识定义见表 E.5。

表 E.5 参变量类对象标识定义

对象标识 OI	接口类 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::= <code>date_time_s</code> 属性 3 (校时模式) ::= <code>enum</code> { 主站授时 (0), 终端精确校时 (1), 北斗/GPS (2), 其他 (255) } 属性 4 (精准校时参数) ::= <code>structure</code> { 最近心跳总个数 <code>unsigned</code> , 最大值剔除个数 <code>unsigned</code> , 最小值剔除个数 <code>unsigned</code> , 通讯延时阈值 <code>unsigned</code> (单位: s), 最少有效个数 <code>unsigned</code> } 方法 127: 广播校时 (参数) 参数::= <code>date_time_s</code>
4001	8	通信地址	属性 2::= <code>octet-string</code>
4002	8	表号	属性 2::= <code>octet-string</code>
4003	8	客户编号	属性 2::= <code>octet-string</code>
4004	8	设备地理位置	属性 2::= <code>structure</code> {

表 E.5 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
4004	8	设备地理位置	经度 structure { 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 { 上电全显时间 unsigned, 背光点亮时间 long-unsigned (按键时背光点亮时间), 显示查看背光点亮时间 long-unsigned, 无电按键屏幕驻留最大时间 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 } 以上时间的单位均为: s。 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
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





表 E.5 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
401A	8	当前套阶梯电价	属性 2 (阶梯参数, 只读) ::= structure { 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 } 阶梯值 ::= double-long-unsigned 阶梯电价 ::= double-long-unsigned 阶梯结算日 ::= structure { 月    unsigned, 日    unsigned, 时    unsigned } 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、时有效则以此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	属性 2 (参数) ::= structure { 月    unsigned, 日    unsigned, 时    unsigned } 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、时有效则以此每月该日时执行阶梯冻结, 当全部无效时不执行阶梯冻结
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 { 报警金额限值 1    double-long-unsigned, 报警金额限值 2    double-long-unsigned } 单位: 元; 换算: -2
401F	8	其他金额限值	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	其他电量限值	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)) 见附录 F

表 E.5 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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 } 版本信息见表 178 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
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) }
4100	8	最大需量周期	属性 2::=unsigned; 单位: min, 换算: 0
4101	8	滑差时间	属性 2::=unsigned; 单位: min, 换算: 0
4102	8	校表脉冲宽度	属性 2 (只读)::=unsigned; 单位: ms; 换算: 0
4103	8	资产管理编码	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

表 E.5 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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 各相电导系数	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 各相电抗系数	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 各相电阻系数	属性 2::=structure { A 相电阻   long (单位: 无; 换算: -3) , 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 各相电纳系数	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 F
4113	8	无功组合方式 1 特征字	属性 2::=bit-string (SIZE (8) ) , 见附录 F
4114	8	无功组合方式 2 特征字	属性 2::=bit-string (SIZE (8) ) , 见附录 F
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	终端广播校时	属性 2 (终端广播校时参数) ::=structure { 终端广播校时启动时间   time, 是否启用                   bool } 属性 3 (终端单地址广播校时参数) ::=structure { 时钟误差阈值               integer (单位: s) ,

表 E.5 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
4204	8	终端广播校时	终端广播校时启动时间    time, 是否启用                    bool } 电能表时钟误差由终端自动计算获得, 前提是采集任务中配置有采集电能表时钟。执行单地址广播校时后生成对应的终端对电能表校时事件
4300	19	电气设备	
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) 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::=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) 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::=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 (参数 1, 参数 2) 参数 1, 热量::=double-long-unsigned 参数 2: 热流量 (水流量)::=double-long-unsigned 用于出厂前机电同步, 仅允许执行一次
4400	20	应用连接	
4401	8	应用连接认证密码	属性 2 (只写)::=visible-string
4500	25	公网通信模块 1	
4501	25	公网通信模块 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	

表 E.5（续）

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
4520	8	公网远程通信多接入 点备用通道	属性 2::=array 备用通道 备用通道::=structure { 运营商代码    unsigned, 网络类型代码  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, 主站通信参数   array structure { IP 地址    octet-string, 端口        long-unsigned } }

E.6 冻结类对象

有关冻结类的对象标识定义见表 E.6。

表 E.6 冻结类对象标识定义

对象标识 OI	接口类 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	
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	

E.7 采集监控类对象

有关采集监控类的对象标识定义见表 E.7。

表 E.7 采集监控类对象标识定义

对象标识 OI	接口类 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> <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）</p> <p>通过配置序号删除配置单元。</p> <p>方法 132: Delete（基本信息）</p> <p>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>
6001	8	采集档案配置单元	<p>属性 2（Acquisition document definition） ::= structure</p> <pre>{     配置序号 long-unsigned,     基本信息 Basic_object,     扩展信息 Extended_object,     附属信息 Annex_object }</pre> <p>Basic_object ::= structure</p> <pre>{     通信地址 TSA,     波特率 enum     {         300bit/s (0), 600bit/s (1), 1200bit/s (2),         2400bit/s (3), 4800bit/s (4), 7200bit/s (5),         9600bit/s (6), 19200bit/s (7), 38400bit/s (8),         57600bit/s (9), 115200bit/s (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),     } }</pre>

表 E.7 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
6001	8	采集档案配置单元	<p>CJ/T 188—2004 (4)</p> <pre> }, 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) } Extended_object ::= structure {     采集器地址  TSA,     资产号      octet-string,     TV 变比     long-unsigned,     TA 变比     long-unsigned } Annex_object ::= array structure {     对象属性描述  OAD,     属性值        Data } </pre>
6002	11	搜表	<p>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</p> <pre> {     启用每天周期搜表标识  bool,     自动更新采集档案      bool,     搜表事件生成标识      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)     } } </pre> <p>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</p> <pre> {     开始时间      time,     搜表时长 (min) long-unsigned } </pre> <p>属性 10 ::= enum</p> <pre> {     空闲 (0), 搜表中 (1) } </pre>



表 E.7 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
6002	11	搜表	} 方法 127: 实时启动搜表 (搜表时长) 搜表时长::=long-unsigned, 单位为 min, 表示搜表持续时间, 0 表示不限时间直至搜表结束。 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 }
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, 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。

表 E.7 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
6012	10	任务配置表	<p>方法 129: Clear (参数) 参数::=NULL 清空任务配置表。</p> <p>方法 130: Update (参数) 参数::=structure {   任务 ID unsigned   状态 enum }</p> <p>更新任务状态</p> <p>方法 131: Update (参数) 参数::=structure {   任务 ID unsigned   参数描述符 OAD,   参数值 Data }</p>
6013	8	任务配置单元	<p>属性 2 (任务配置单元)::=structure {   任务 ID unsigned,   执行频率 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, }</p> <p>当方案类型为脚本时, 方案编号为脚本 ID</p> <p>任务运行时段::=structrue {   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 }</p> <p>时段::=structure {</p>

表 E.7（续）

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义																		
6013	8	任务配置单元	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, 表计集合 MS, 存储时标选择 enum { 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) } } 采集方式数据格式如下: <table><tr><th>采集类型</th><th>采集内容</th><th>表示</th></tr><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></table> RetryMetering ::= structure	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																			

表 E.7 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义															
6015	8	普通采集方案	<pre>{     数据时标间隔      TI,     补抄周期（上 N个） long-unsigned }</pre>															
6016	11	事件采集方案集	属性 2::=array 事件采集方案 方法 127: Add (array 事件采集方案) 添加或更新一组事件采集方案。 方法 128: Delete (array 方案编号) 删除一组事件采集方案。 方法 129: Clear (参数) 参数::=NULL 清空事件采集方案集。 方法 130: UpdateReportFlag(参数) 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><tr><th>采集类型</th><th>采集内容</th><th>表示</th></tr><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></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      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)															

表 E.7 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
6018	11	透明方案集	参数::=structure { 方案编号  unsigned 通信地址集合  array TSA } 删除一个方案的一组方案内容。 方法 130: Delete (array 方案编号) 删除一组透明方案。 方法 131: Clear (参数) 参数::=NULL 清空透明方案集
6019	8	透明方案	属性 2 (透明方案)::=structure { 方案编号      unsigned, 方案内容集  array 方案内容, 存储深度     long-unsigned } 方案内容::=structure { 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, 方案控制标识      structure, 方案报文集        array 方案报文 } 方案控制标识::=structure { 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 (s)        long-unsigned, 结果比对标识          enum{不比对 (0), 比 (1), 比对上报 (2)}, 结果比对参数          structure } 方案报文::=structure { 报文序号  unsigned, 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, }

表 E.7 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义									
601B	8	一个透明方案结果	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, 最大上报次数      unsigned, 上报内容      strcuture { 类型    unsigned, 数据    Data } } 上报内容: <table><tr><td>上报类型</td><td>上报内容</td><td>表示</td></tr><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></table> RecordData::=structure { 主对象属性描述符      OAD, 记录型对象属性描述符    RCSD, 行选择      RSD }	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	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)} }									

表 E.7 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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	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, 采集总数量           long-unsigned, 采集成功数量       long-unsigned, 已发送报文条数      long-unsigned, 已接收报文条数      long-unsigned } 说明: 采集成功数量、已发送报文条数、已接收报文条数, 每次任务执行先清零
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

### E.8 集合类对象

有关集合类的对象标识定义见表 E.8。

表 E.8 集合类对象标识定义

对象标识 OI	接口类 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) 删除文件。 文件名::=visible-string 方法 130: ReadFile (参数) 参数::=structure { 文件名     visible-string, 偏移       double-long-unsigned } 应答::=octet-string

表 E.8 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
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: 单位为字节
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) 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

## E.9 控制类对象

有关控制类的对象标识定义见表 E.9。



表 E.9 控制类对象标识定义

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
8000	8	遥控	<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：min，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”为处于告警状态，置“0”为未处于告警状态。</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输出状态，置“1”为跳闸命令，置“0”为合闸命令。</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：min；换算：0），     限电时间    long-unsigned（单位：min；换算：0。值为“0”表示永久限电），     自动合闸    bool（True：自动合闸；False：非自动合闸） }</pre> <p>方法 130：合闸（参数） 参数 ::= array structure</p> <pre>{     继电器    OAD,     命令    enum{合闸允许（0），直接合闸（1）} }</pre> <p>方法 131：电能表明文合闸（参数） 参数 ::= array structure</p> <pre>{     继电器    OAD,     命令    enum{合闸允许（0），直接合闸（1）},     密码    visible-string }</pre>
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 允许与主站最大无通信时长（min），0 表示不自动保电。</p> <p>属性 4 ::= long-unsigned 上电自动保电时长（min），“0”表示上电不自动保电。</p> <p>属性 5 ::= array 自动保电时段 自动保电时段 ::= structure</p> <pre>{     起始时间（时）    unsigned,     结束时间（时）    unsigned }</pre> <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>

表 E.9 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
8001	8	保电	<p>参数::=NULL 用于投入保电状态，即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) 参数::=NULL 用于解除保电状态。</p> <p>方法 129: 解除自动保电 (催费告警参数) 参数::=NULL 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)} 属性 3 (催费告警参数, 只读) ::=structure {     告警时段  octet-string(SIZE(3)),     告警信息  visible-string(SIZE(1~200)) } 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。</p> <p>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</p>
8003	11	一般中文信息	<p>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 (200)) } 方法 127: 添加信息 序号::=unsigned 发布时间::=date_time_s 信息内容::=visible-string (SIZE (200)) 方法 128: 删除信息 (序号) “序号” 参见方法 127</p>
8004	11	重要中文信息	<p>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</p>
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	<p>属性 2 (配置参数) ::=array unsigned 终端功控时段单元格格式见表 E.10</p>
8102	8	功控告警时间	<p>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: min)</p>

表 E.9 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
8103	13	时段功控	<p>属性 2 (控制方案集) ::= array 时段功控配置单元</p> <p>方法 127: 时段功控方案切换 (参数)</p> <p>参数 ::= structure</p> <pre>{     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     } }</pre> <p>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”为有效, 置“0”为无效。</p> <p>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, 其他值无效</p>
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	<p>属性 2: 不可访问</p> <p>方法 127 投入 (参数)</p> <p>参数 ::= structure</p> <pre>{     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: min),         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: min),         当前功率下浮控的控制时间  unsigned (单位: 0.5h),         当前功率下浮控第 1 轮告警时间  unsigned (单位: min),         当前功率下浮控第 2 轮告警时间  unsigned (单位: min),         当前功率下浮控第 3 轮告警时间  unsigned (单位: min),         当前功率下浮控第 4 轮告警时间  unsigned (单位: min)     } }</pre> <p>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</p>
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	<p>属性 2 ::= structure</p> <pre>{     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,     第三套定值  PowerCtrlParam,     时段功控定值浮动系数  integer (单位: %) }</pre> <p>方案标识: bit0~bit2 按顺序对位表示第 1~3 套定值, 置“1”为有效, 置“0”为无效。</p> <p>PowerCtrlParam ::= structure</p>

表 E.9 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
8109	8	时段功控配置单元	<pre> {     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) } </pre> <p>时段号: bit0~bit7 按顺序对位表示第 1~8 时段, 置“1”为有效, 置“0”为无效</p>
810A	8	厂休控配置单元	<pre> 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: min) ,     每周限电日      bit-string (SIZE (8)) } </pre> <p>每周限电日: D1~D7 表示星期一~星期日, D0=0</p>
810B	8	营业报停控配置单元	<pre> 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) } </pre>
810C	8	购电控配置单元	<pre> 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)} ,     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)} } </pre>
810D	8	月电控配置单元	<pre> 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: %) } </pre>
810E	8	控制对象	
810F	8	跳闸轮次	
8110	8	电控定值	

终端功控时段数据单元格式见表 E.10。

表 E.10 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
<p>注 1：每 0.5h 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。</p> <p>注 2：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。</p>								

E.10 文件传输类对象

有关文件传输类的对象标识定义见表 E.11。

表 E.11 文件传输类对象标识定义

对象标识 OI	接口类 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	<p>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。</p> <p>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，单位为 B</p>
F001	18	文件分块传输管理	<p>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=0~总传输块数-1）。</p> <p>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 }</p> <p>文件信息数据结构见表 176 中属性 2。</p> <p>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 }</p> <p>方法 8：写文件（参数） 参数 ::= structure</p>

表 E.11 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
F001	18	文件分块传输管理	<pre> {     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 } 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: 对设备的软件进行比对, 命令中 CPU 编号的 bit0~bit2 有效, 其他保留。该字节默认为“00”, 如设备内部存在多个 CPU, 主 CPU 编号为 0, 其他自行编号, 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。</p> <p>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 G</p>
F002	18	文件扩展传输管理	<pre> 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 } </pre> <p>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。</p> <p>方法 7: 从服务器下载 (文件信息, 协议类型)</p> <pre> 协议类型::=enum {     telnet+zmodem 协议  (0),     ftp 协议        (1),     sftp 协议        (2), </pre>

表 E.11 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
F002	18	文件扩展传输管理	<p>http 协议 (3), https 协议 (4)</p> <p>}</p> <p>以客户机模式主动连接指定远程服务器下载文件, 并通过“命令结果”反馈执行情况。目标文件不存在则创建。</p> <p>方法 8: 上传到服务器 (参数)</p> <p>参数 ::= structure</p> <p>{</p> <p>文件信息 structure</p> <p>协议类型 enum</p> <p>}</p> <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</p>

E.11 ESAM 接口类对象

有关 ESAM 接口类的对象标识定义见表 E.12。

表 E.12 ESAM 接口类对象标识定义

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
F100	21	ESAM	
F101	8	安全模式参数	<p>属性 2 (安全模式选择) ::= enum</p> <p>{</p> <p>不启用安全模式参数 (0),</p> <p>启用安全模式参数 (1)</p> <p>}</p> <p>属性 3 (显式安全模式参数) ::= array 安全模式参数</p> <p>安全模式参数 ::= structure</p> <p>{</p> <p>对象标识 OI,</p> <p>安全模式 long-unsigned</p> <p>}</p> <p>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 H。</p> <p>属性 4 (SAL 安全应用数据链路层参数) ::= enum</p> <p>{</p> <p>不启用 SAL (0),</p> <p>启用 SAL (1)</p> <p>}</p> <p>方法 1: 复位 (参数)</p> <p>参数 ::= integer (0)</p> <p>复位时, 清空属性 3。</p> <p>方法 127: 增加显式安全对象 (参数)</p> <p>参数 ::= structure</p> <p>{</p> <p>对象标识 OI,</p> <p>权限 long-unsigned</p> <p>}</p>

表 E.12 (续)

对象标识 OI	接口类 IC	对象名称	实例的对象属性及方法定义
F101	8	安全模式参数	方法 128: 删除显式安全模式参数 (对象标识) 对象标识::=OI 方法 129: 批量增加显式安全模式参数 (array 安全对象) 安全对象::=structure { 对象标识     OI, 安全模式     long-unsigned }

## E.12 输入输出设备类对象

有关输入输出设备类对象标识定义见表 E.13。

表 E.13 输入输出设备类对象标识定义

对象标识 OB	接口类 IC	对象名称	实例的对象属性及方法定义
F200	22	RS-232	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{上行通信 (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 }
F201	22	RS-485	
F202	22	红外	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 }
F203	22	开关量输入	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。



表 E.13 (续)

对象标识 OB	接口类 IC	对象名称	实例的对象属性及方法定义
F203	22	开关量输入	属性 4::=structure { 开关量接入标识 bit-string (SIZE (8)) ( 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。 ), 开关量属性标识 bit-string (SIZE (8)) ( bit0~bit7 按顺序对位表示第 1~8 路状态量输入, 置“1”动 合触点, 置“0”为动断触点。 ) }
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, 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),           }

表 E.13 (续)

对象标识 OB	接口类 IC	对象名称	实例的对象属性及方法定义
F207	22	多功能端子	需量周期 (1), 时段投切 (2) } 方法 127: 修改工作模式 (参数) 参数::=structure { 路号 OAD, 端子功能 enum }
F208	22	交流采样接口	属性 2 (设备对象列表, 只读) ::=array 交流采样单元 交流采样单元::=structure { 交流采样描述符 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 (更新周期) ::= TI 属性 7 (网络拓扑信息, 只读) ::=array 网络拓扑单元 属性 8 (多网信息, 只读) ::=structure { 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)) } 属性 9 (宽带载波频段序号) ::=unsigned 方法 127: 透明转发 (参数) 参数::=structure { 通信地址 TSA, 接收等待报文超时时间 (秒) long-unsigned, 透明转发命令 octet-string } 返回结果::=octet-string 方法 128: 配置端口参数 (参数) 参数::=structure { 端口号 OAD, 通信参数 COMDCB }

表 E.13（续）

对象标识 OB	接口类 IC	对象名称	实例的对象属性及方法定义
F20A	22	脉冲输入设备	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
F20B	22	蓝牙	属性 2（设备对象列表，只读） ::= array 蓝牙模块 蓝牙模块 ::= structure { 端口描述符     visible-string, 通信参数       COMDCB } 方法 127：配置端口（参数） 参数 ::= structure { 端口号     OAD, 通信参数   COMDCB }
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 （单位为 dB $\mu$ V）
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 从节点响应时长   long-unsigned（单位：s） 从节点最近一次通信成功时间   date_time_s }
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）

E.13 显示类对象

有关显示类的对象标识定义见表 E.14。

表 E.14 显示类对象标识定义

对象标识 OI	接口类 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	

附 录 F  
(资料性附录)  
状态字、特征字、模式字

**F.1 电能表运行状态字 1**

电能表运行状态字 1 的格式见表 F.1。

**表 F.1 电能表运行状态字 1 格式**

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
保留	保留	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

**F.2 电能表运行状态字 2**

电能表运行状态字 2 的格式见表 F.2。

**表 F.2 电能表运行状态字 2 格式**

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 为正向，1 为反向。							

**F.3 电能表运行状态字 3（操作类）**

电能表运行状态字 3（操作类）的格式见表 F.3。

**表 F.3 电能表运行状态字 3（操作类）格式**

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

表 F.3 (续)

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)	
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。 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”。 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”。 注 4: 预跳闸报警状态 (bit7): 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”。 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)。 注 6: bit0、bit5、bit10、bit11 保留, 置“0”。							

## F.4 电能表运行状态字 4 (A 相故障状态)

电能表运行状态字 4 的格式见表 F.4。

表 F.4 电能表运行状态字 4 格式

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 无此类故障, 1 当前发生此类故障。							

## F.5 电能表运行状态字 5 (B 相故障状态)

电能表运行状态字 5 的格式见表 F.5。

表 F.5 电能表运行状态字 5 格式

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 无此类故障, 1 当前发生此类故障。							

## F.6 电能表运行状态字 6 (C 相故障状态)

电能表运行状态字 6 的格式见表 F.6。

表 F.6 电能表运行状态字 6 格式

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

表 F.6 (续)

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 无此类故障，1 当前发生此类故障。							

#### F.7 电能表运行状态字 7 (合相故障状态)

电能表运行状态字 7 (合相故障状态) 的格式见表 F.7。

表 F.7 电能表运行状态字 7 格式

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 无此类故障，1 当前发生此类故障。							

#### F.8 有功组合方式特征字

有功组合方式特征字的格式见表 F.8。

表 F.8 有功组合方式特征字格式

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)

#### F.9 无功组合方式 1、2 特征字

无功组合方式 1、2 特征字的格式见表 F.9。

表 F.9 无功组合方式 1、2 特征字格式

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)

#### F.10 周休日特征字

周休日特征字的格式见表 F.10。

表 F.10 周休日特征字格式

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 休息，1 工作。							

**F.11 插卡状态字**

插卡状态字的格式见表 F.11。

**表 F.11 插卡状态字格式**

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: bit1、bit0 在插卡操作后置相应状态, 读取和上电后置未知。

**F.12 控制命令执行状态字**

控制命令执行状态字的格式见表 F.12。

**表 F.12 控制命令执行状态字格式**

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

**F.13 控制命令错误状态字**

控制命令错误状态字的格式见表 F.13。

**表 F.13 控制命令错误状态字格式**

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, 对 应的错误 不会发生	安全认证 超时	密码错误/ 未授权	保留, 对 应的错误 不会发生	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。

**F.14 电能表跟随上报状态字**

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	时钟电池电 压低	保留	ESAM 错误	保留



表 F.14（续）

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	保留	保留	保留	保留	保留	保留
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：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。</p> <p>注 2：确认电能表跟随上报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，则电能表清零电能表跟随上报状态字中对应位。</p>							

## 附 录 G

### （规范性附录）

### 安 全 认 证 说 明

#### G.1 比对加密方式

##### G.1.1 补位规则

如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。

##### G.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 8B，从比对因子起始地址开始，在程序存储器中取 16B。将 16B 分为前 8B 和后 8B 两个数据块 (Data1, Data2)，然后对数据块进行处理 ( $\text{Data1} \oplus \text{Data2} = \text{Data3}$ ，其中  $\oplus$  代表异或运算符)，得到异或后数据块 Data3 作为比对因子；
- b) 提取随机数：共 16B，从比对数据起始地址开始取 64B，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( $\text{Data4} \oplus \text{Data5} \oplus \text{Data6} \oplus \text{Data7} = \text{Data8}$ ，其中  $\oplus$  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( $\text{Data9} \oplus \text{Data10} \oplus \text{Data11} \oplus \text{Data12} = \text{Data13}$ ，其中  $\oplus$  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。

##### G.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 ( $\text{Data1} \oplus \text{Data2} \oplus \text{Data3} \oplus \text{Data4} = \text{Data5}$ ，其中  $\oplus$  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 ( $\text{Data6} \oplus \text{Data7} \oplus \text{Data8} \oplus \text{Data9} = \text{Data10}$ ，其中  $\oplus$  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。

#### G.2 安全方式说明

安全方式说明如下：

- 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。

附 录 H  
(资料性附录)  
安 全 模 式 参 数

**H.1 安全模式参数**

安全模式参数设置值的定义见表 H.1。

**表 H.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
长度	2B	2B			

安全模式定义见表 H.2。

**表 H.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

**H.2 默认安全模式参数**

默认安全模式参数的定义见表 H.3。

**表 H.3 默认安全模式参数定义**

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	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			
0ZZZ	当前电能	√												√		
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	√					√							√		

表 H.3 (续)

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	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			
4003	客户编号		√				√									
400A	备用套分时费率切换时间		√						√							
400B	备用套阶梯切换时间		√						√							
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	采集监控	√				√				√				√	√	√
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	√				√				√				√	√	√

表 H.3（续）

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	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			
FFZZ	自定义	√				√				√				√	√	√
<p>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。</p> <p>注 2：其他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。</p> <p>注 3：除以上表格中规定外，其他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。</p> <p>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。</p> <p>注 5：设置基表远程通信模块的信号强度时，不需硬件配合，不需密码验证。</p>																

中 华 人 民 共 和 国  
电 力 行 业 标 准  
电能信息采集与管理系统  
第 4-5 部分：通信协议——面向  
对象的数据交换协议  
DL/T 698.45—2017  
代替 DL/T 698—1999

\*

中国电力出版社出版、发行  
(北京市东城区北京站西街 19 号 100005 <http://www.cepp.sgcc.com.cn>)  
北京传奇佳彩印刷有限公司印刷

\*

2018 年 10 月第一版 2018 年 10 月北京第一次印刷  
880 毫米×1230 毫米 16 开本 13 印张 395 千字  
印数 001—100 册

\*

统一书号 155198·831

版 权 专 有 侵 权 必 究  
本书如有印装质量问题，我社发行部负责退换



中国电力出版社官方微信



电力标准信息微信

为您提供 **最及时、最准确、最权威** 的电力标准信息



155198.831