ISO 15765-2（2004）

道路车辆——控制局域网络诊断——

第2部分：

网络层服务

ISO 15765协议，定义了“道路车辆——局域网控制器（CAN）诊断”，包含如下几个部分：

——第一部分：总论。

——第二部分：网络层服务。

——第三部分：统一诊断服务具体的执行

——第四部分：排放相关系统的要求

**概述**

ISO15765-2的协议，定义车载诊断系统网络层要求，提供在CAN数据链路层（ISO11898定义）上运行。虽说它最初设计是用在诊断系统上的，它同样适用于其它需要网络层协议的CAN通信系统上。

为了达到诊断通信要求，ISO15765协议是基于ISO/IEC 7498和ISO/IEC 10731的开放互联系统基本参考模型建立的。该模型将通信系统分为七层，ISO15765协议映射到该模型上，分层如下：

——统一的诊断服务（第7层），在ISO 15765-3中定义。

——网络层（第3层），在ISO 15765-2中定义。

——CAN服务层（第1，2层），在ISO 11898中定义。

如表1所示，

表1——对应OSI分层，扩展的及法规要求的OBD诊断规定

|  |  |  |
| --- | --- | --- |
| 开放互联系统  （OSI）分层 | 汽车生产商扩展的诊断服务 | 法规要求的车载诊断系统  （OBD） |
| 诊断应用 | 用户定义 | ISO 15031-5 |
| 应用层 | ISO 15765-3 | ISO 15031-5 |
| 表示层 | N/A | N/A |
| 会话层 | ISO 15765-3 | N/A |
| 传输层 | N/A | N/A |
| 网络层 | ISO 15765-2 | ISO 15765-4 |
| 数据链路层 | ISO 11898-1 | ISO 15765-4 |
| 物理层 | 用户定义 | ISO 15765-4 |

应用层服务是由ISO 15765-3按照ISO 14229-1和ISO 15031-5建立的诊断服务制定的，但ISO 15765-3协议不仅适用于上述的诊断服务项，也适用于大多数的国际标准及汽车生产商规定的诊断服务。

网络层服务由该部分的ISO 15765协议定义，并独立于物理层上的操作，物理层仅仅是在法规的OBD上有规定。

**道路车辆——控制器局域网（CAN）的诊断——**

**第二部分：**

**网络层**

1. **范围**

这部分ISO 15765协议描述了在ISO 11898定义的控制其局域网中裁剪的网络协议，用于满足基于CAN的车载网络系统。它是按照ISO 14229-1和ISO 15031-5建立的诊断服务制定的，但该部分协议不仅适用于上述的诊断服务项，还适用于车载内部其它的网络通信。该协议描述的是未经最后确认的的通信。

1. **参考的标准**

下述的参考文档对于该文档的应用是必不可少的。

ISO 11898-1，道路车辆——控制器局域网（CAN）——第一部分——数据链路层及物理信号层

ISO/IEC 7498(所有部分)，技术信息——开放互联系统——基本参考模型

1. **术语，定义和缩略词**

为编撰该文档目的，这些术语和定义已在ISO 7498中给出，以下缩略词术语同样适用。

BS 数据块大小

CF 连续帧

confirm 确认服务

ECU 电子控制单元

FC 流控制

FF 首帧

FF\_DL 首帧数据长度

FS 流状态

indication 指示服务

MType 信息类型

N\_AE 网络地址扩展

N\_AI 地址信息

N\_Ar 网络层时间参数：Ar

N\_As 网络层时间参数：As

N\_Br 网络层时间参数：Br

N\_Bs 网络层时间参数：Bs

N\_ChangeParameter 网络层服务项名称

N\_Cr 网络层时间参数：Cr

N\_Cs 网络层时间参数：Cs

N\_Data 网络数据

N\_PCI 网络协议控制信息

N\_PCItype 网络协议控制信息类型

N\_PDU 网络协议数据单元

N\_SA 网络源地址

N\_SDU 网络服务数据单元

N\_TA 网络目标地址

N\_TAtype 网络目标地址类型

N\_USData 网络层无应答的数据段传输服务项名称

NWL 网络层

request 应答服务

r 接收者

s 发送者

SF 单帧

SF\_DL 单帧数据长度

SN 顺序号

STmin 间隔最短时长

1. **网络层总览**
   1. **概述**

该项主要描述网络层总体的功能。该部分的ISO 15765协议定义了未最后确认的网络层通信协议。该协议用于网络节点之间数据交互，例如从一个ECU到另一个ECU，或外部诊断设备和一个ECU之间的通信。如果要传送的数据超过了单个的CAN帧长度，则需要提供拆分的方法。

为描述网络层的功能，它提供给高层的服务项及内部操作必须予以研究。

* 1. **网络层提供给高层的服务项**

该服务项接口定义了一些由网络层提供使用服务项，例如，数据发送、数据接收及协议参数设置。

已定义了两种类型的服务：

1. **通信服务项**

以下定义的服务项，使发送者最多能发送4095个字节的数据。

1. N\_USData.request

该服务项用于请求发送数据。如果有必要的话，网络层拆分这些数据。

1. N\_USData\_FF.indication

该服务项用于通知上层被拆分的信息的首帧的接收。

1. N\_USData.Indication

该服务项用于提供接收的数据至上层。

1. N\_USData.comfirm

该服务项用于确认应答给上层，表示请求服务项已经被执行（成功执行或不成功执行）。

1. **协议参数设置服务项**

以下定义的服务项，使之能够对协议参数动态设置。

1. N\_ChangeParamter.request

该项服务用于对特定内部参数的动态设置的请求

1. N\_ChangeParameter.comfirm
2. 该服务项用于确认应答给上层，表示修改协议特定项的请求已经被执行（成功执行或不成功执行）。

**4.3 网络层的内部操作**

网络层的内部操作为实现对等实体间的通信提供了分段、重组、数据传输流控制方法。网络层主要的任务是传递一帧或大于一帧的数据信息。超过一帧的信息被分成多个部分，每一个部分都以一个CAN帧的形式被发送。

图1显示的是未被拆分的信息的传送的例子。

图2显示的是被拆分的信息传送的例子

流控制用来使发送端适应接收端网络层的接收能力。该流控制策略同样适用于诊断网关和通信子网。

1. **网络层服务项**
   1. **总览**

所有的网络层服务项有统一的结构。为了定义这些服务项，三类主要的服务项说明如下：

——请求服务，被更高的通信层或应用层使用，用于向网络层传递控制信息及要发送的数据；

——指示服务，被网络层使用，用于向更高通信层或应用层传递状态信息及接收到的数据；

——确认服务，被网络层使用，用于向更高通信层或应用层传递状态信息。

这些服务说明没有指定具体的应用程序接口，而只是一些独立于具体实施的主要服务项。

所有的网络层服务项有统一的结构形式，服务项写成如下的形式：

service\_name.type (

parameter A,

parameter B,

parameter C,

…

)

这里，“service\_name”是指服务项名称，例如，N\_SDU，“type”指示了服务项的类型，“parameter A,parameter B,parameter C,…”则是N\_SDU服务项传递的值。

服务项定义了如何使服务的使用者(例如，诊断应用层)如何与服务的提供者（例如，网络层）协同运行。以下服务项已在国际标准中说明，请求，指示和确认。

——使用请求服务项（service\_name.request），服务使用者向服务提供者请求一项服务。

——使用指示服务项(service\_name.indication)，服务提供者通知服务使用者网络层的一个内部事件或者一个对等实体的服务使用者的服务请求。

——通过确认服务项（service\_name.confirm），服务提供者通知服务的使用者，之前服务使用者请求服务的结果。

* 1. **网络层服务说明**
     1. N\_USData.request

该请求服务项是请求传递<MessageData>数据及<Length>字节数，从发送者到到对等实体接收者，通过在N\_SA,N\_TA,N\_TAtype及N\_AE中的地址信息确认。（参看5.3对参数的定义）。

N\_USData.request服务项每次被启动，网络层应当通过一条N\_USData.confirm服务通知服务使用者信息传递的完成情况。（成功或失败）

N\_USData.request （

Mtype

N\_SA

N\_TA

N\_TAtype

N\_AE（可选的）

<MessageData>

<Length>

)

* + 1. N\_USData.confirm

N\_USData.confirm服务项由网络层发送，该服务项用于确定N\_USData.request服务的完成情况，通过在N\_SA,N\_TA,N\_TAtype及N\_AE中的地址信息确认。参数<N\_Result>提供请求服务项的状态。（参看5.3对参数的定义）

N\_USData. confirm（

Mtype

N\_SA

N\_TA

N\_TAtype

N\_AE（可选的）

<N\_Result>

)

* + 1. N\_USData\_FF.indication

N\_USData\_FF. indication服务项由网络层发送。该服务项用于通知相邻上层接收到对等实体首帧数据已经到了。通过在N\_SA,N\_TA,N\_TAtype及N\_AE中的地址信息确认。（参看5.3对参数的定义）这个指示项发生在接收到拆分数据首帧的时刻。

N\_USData\_FF.indication （

Mtype

N\_SA

N\_TA

N\_TAtype

N\_AE（可选的）

<Length>

)

N\_USData\_FF. indication指示服务项发送完，网络层应当总是紧跟着发送一个N\_UDSData.indication服务项，指示信息接收的完成情况。（成功或失败）

N\_USData\_FF. indication指示服务项应当至友网络层发送指示信息段的首帧是否被正确接收。

如果网络层监测到首帧中任何类型的错误，该信息应当被网络层忽略，并且N\_USData\_FF. indication指示服务项不应当被发送至相邻的上层。

如果网络层接收到首帧中数据长度项的值(FF\_DL)大于接收者缓冲区的数据，这应当被认为是一个错误的条件并且N\_USData\_FF. indication指示服务项不应当被发送至相邻的上层。

* + 1. N\_USData.indication

N\_USData.indication服务项由网络层发送。该服务项指示<N\_Result>事件并传递<Length>字节数的<MessageData>至相邻的上层。这些信息通过同等实体间通过存放于 N\_SA,N\_TA,N\_TAtype及N\_AE中标识的地址信息接收过来的。

当<N\_Result>值为N\_OK时，<MessageDate>及<Length>参数信息才有效。

N\_USData.indication （

Mtype

N\_SA

N\_TA

N\_TAtype

N\_AE（可选的）

<MessageData>

<Length>

<N\_Result>

)

N\_USData.indication服务项是在接收到单帧（SF）信息或是指示拆分信息接收的完成时发送。

如果网络层检查到单帧中任何类型的错误，该条单帧信息应当被忽略并且N\_USData\_FF. indication指示服务项不应当被发送至相邻的上层。

* + 1. N\_ChangeParameters.request

该服务项用于请求本地实体内部参数的修改。<Parameter\_Value>参数值分配给<Parameter>参数（参看5.3对参数的定义）。

对参数总是可以修改的。特殊情况是在应用层接收到首帧的指示服务项（N\_USData\_FF.indication）到接收（N\_USData.indication）服务项之间的时刻。

N\_ChangeParameters.request （

Mtype

N\_SA

N\_TA

N\_TAtype

N\_AE（可选的）

<Parameter>

<Parameter\_Value>

)

这是一个可选服务项，可被固定的参数值实施代替。

* + 1. N\_ChangeParameters.confirm

该服务项用于确认N\_ChangeParameter.Confirmation运用信息的完成情况，这信息通过在N\_SA,N\_TA,N\_TAtype及N\_AE中的地址信息标识。

N\_ChangeParameter.confirm （

Mtype

N\_SA

N\_TA

N\_TAtype

N\_AE（可选的）

<Parameter>

<Result\_ChangeParameter>

)

5.3 服务项数据单元说明

5.3.1 Mtype, Message type

类型：枚举类型

范围：诊断，远程诊断

描述：参数Mtype用于确定服务相中信息参数的类型及范围。该部分的ISO 15765协议指定了两个值标识这个参数。文档使用者可通过指定其它的类型，也可通过文档中网络层使用的其它地址信息参数的组合来扩展这些值的范围。每新定义的一套地址信息，Mtype应当赋予新值，标识新的地址信息。

——如果Mtype = 诊断，N\_AI地址信息应当包含参数N\_SA,N\_TA,和N\_TAtype。

——如果Mtype = 远程诊断，N\_AI地址信息应当包含参数N\_SA,N\_TA,和N\_TAtype，和N\_AE。

5.3.2 N\_AI,地址信息

5.3.2.1 N\_AI描述

该参数指的是地址信息。总的来说，N\_AI参数用于确定信息发送者和接收者的源地址（N\_SA），目标地址（N\_TA），也包含确定(N\_TAtype)和可选择地址扩展（N\_AE）的通信模式。

5.3.2.2 N\_SA网络源地址

类型：1字节的无符号整数

范围：00-FF 16进制

描述：N\_SA参数代表发送者网络层实体

5.3.2.3 N\_TA，网络目标地址

类型：1字节的无符号整数

范围：00-FF 16进制

描述：N\_SA参数代表接收者网络层实体

5.3.2.4 N\_TAtype,网络目标地址类型

类型：枚举类型

范围：物理的，功能的

描述：N\_TAtype参数是对N\_TA参数的扩展。它被网络层对等实体使用，代表通信模式。两种通信模式说明如下：1对1的通信，称为物理地址，1对多的通信称为功能地址。

——物理地址（1对1通信）网络层所有类型的信息都支持。

——功能地址（1对多通信）仅仅对单帧的通信支持。

5.3.2.5 N\_AE,网络地址扩展

类型：1字节的无符号整数

范围：00-FF 16进制

描述：N\_AE参数用于在大的网络上扩展现行的地址范围，用于子网中发送与接收网络层实体而不是本地网的通信。若Mtype设置为远程诊断时，N\_AE仅仅是地址信息的一部分。

5.3.3<Length>

类型： 12个bit位

范围： 1-4095

描述：该参数包含要发送或接收的数据长度。

5.3.4<MessageData>

类型：字符串

范围：不固定

描述：该参数包含与上层实体所有交互的数据

5.3.5<Parameter>

类型：枚举类型

范围：STmin, BS

描述：该参数确定网络层的参数

5.3.6<Parameter\_Value>

类型：1字节无符号整数

范围：0-255

描述：该参数分配给协议参数<Parameter>作为指示服务。

5.3.7<N\_Result>

类型：枚举类型

范围：N\_OK, N\_TIMEOUT\_A, N\_TIMEOUT\_Bs, N\_TIMEOUT\_Cr, N\_WRONG\_SN,N\_INVALID\_FS, N\_UNEXP\_PDU, N\_WFT\_OVRN, N\_BUFFER\_OVFLW, N\_ERROR

描述：该参数包含服务项执行的结果状态。如果同时产生了两个或以上的错误，网络层应该使用下列错误指示中首先找到的参数值，发送给高层。

——N\_OK

该值表示服务执行完全正确；它可同时由发送者和接收者发送至服务的使用者。

——N\_TIMEOUT\_A

该值在定时器N\_Ar/N\_As超过了定时值N\_Asmax/N\_Armax，发送给服务的使用者；它可同时由发送者和接收者发送至服务的使用者。

——N\_TIMEOUT\_Bs

该值在定时器N\_Bs超过了定时值N\_Bsmax，发送给服务的使用者；它仅能由发送者发送至服务的使用者。

——N\_TIMEOUT\_Cr

该值在定时器N\_Bs超过了定时值N\_Crmax，发送给服务的使用者；它仅能由接收者发送至服务的使用者。

——N\_WRONG\_SN

该值在接收到意外的连续的数值（PCI.SN）时被发送至服务使用者；它仅能由接收者发送至服务的使用者。

——N\_INVALID\_FS

该值在从流控（FC）N\_PDU接收到无效的或未知的流状态值时发送至服务的使用者；它仅能由发送者发送至服务的使用者。

——N\_UNEXP\_PDU

该值在接收到未知协议数据单元时发送给服务使用者，它仅能由接收者发送至服务的使用者。

——N\_WFT\_OVRN

该值在接收到流控WAIT帧超过最大计数N\_WFTmax时发送至服务使用者。

——N\_BUFFER\_OVFLW

该值在接收到流控（FC）N\_PDU状态Flow = OVFLW时发送给服务的使用者。它用于指示接收者缓冲区无法存储首帧中数据长度（FF\_DL），因此，该拆分数据的传递被丢弃。它仅能由发送者发送至服务的使用者。

——N\_ERROR

这是一个默认的错误值。它是当检测到网络层错误并且没有其它更好的参数描述该项错误时使用发送到服务使用者。它可同时由发送者和接收者发送至服务的使用者。

5.3.8 <Result\_ChangeParameter>

类型： 枚举类型

范围：N\_OK, N\_RX\_ON, N\_WRONG\_PARAMETER, N\_WRONG\_VALUE

描述：该参数包含服务执行的结果状态信息

——N\_OK

该值表示服务执行完全正确；它可同时由发送者和接收者发送至服务的使用者。

——N\_RX\_ON

该值发送给服务使用者指示虽然<AI>标识的信息接收到了，但服务没有执行。它仅能由接收者发送至服务的使用者。

——N\_WRONG\_PARAMETER

该值发送给服务的使用者表示由于未定义的<Parameter>，服务没有执行；它可同时由发送者和接收者发送至服务的使用者。

——N\_WRONG\_VALUE

改制发送给服务的使用者表示由于<Parameter\_Value>超出范围，服务没有执行，它可同时由发送者和接收者发送至服务的使用者。

1. 网络层协议
   1. 协议功能

网络层协议协议有如下功能：

1. 发送/接收最多4095个字节的数据信息
2. 报告发送/接收完成状态。
   1. 单帧发送

（扩展及混合地址情况下）最多发送6字节或（正常地址情况下）7字节数据，按照N\_PDU格式发送信息，称为单帧（SF）见图3.

最多6或7字节的接收通过N\_PDU制式，

图3 未拆分数据的例子

* 1. 多帧发送

长信息的发送通过拆分信息并通过多个N\_PDU发送的形式。长信息的接收通过接受多个N\_PDU并通过重组这些接受的数据。这多个N\_PDU包括首帧（信息中第一个N\_PDU）及连续帧（剩下的所有N\_PDU）。

多N\_PDU信息接收者有条件按照它自己的接收能力通过使用流控协议数据单元（FC N\_PDU）的流控机制调整传输流量。

——一个首帧协议数据单元（FF N\_PDU），包括（扩展及混合地址情况下）5字节或（正常地址情况下）6字节数据。

——一个多更多连续帧协议数据单元（CF N\_PDU），包括6或7字节数据。CF N\_PDU包括剩下的字节数据，因此可以少于6或7字节的长度。

图4显示的是发送端拆分及接收端重组信息的例子

说明：接收方发送的FC N\_PDU用于应答FF N\_PDU，这在图中没有显示。

图4—拆分与重组

信息的长度在首帧（FF N\_PDU）中被发送。所有的连续帧（CF N\_PDU）都被发送方编号，提供接收方以同样的顺序重组这些信息。

接收方通过流控机制（见图5）通知发送方接收能力。由于不同的节点有不同的接收能力，发送方发送的流控通知接收方，接收方才能调整以适应发送方接收能力。

该接收能力如下定义：

——块大小（BS）：在授权继续发送其余的N\_PDU之前，接收方允许发送方最大的N\_PDU个数。

——间隔最短时长（STmin）：发送方在发送两个连续帧间隔等待的最短时间。

图5 流控机制

除了最后一个所有的块都包含BS N\_PDUs。最后一个块包含剩余的N\_PDUs（<=BS）。

每一个发送者/接收者等待对方的N\_PDU时，一个超时机制可用来检查发送失败。

通过 FC N\_PDUs,接收者有权控制CF N\_PDUs,用于延时对方的发送及当拆分数据超过接收者缓冲区的时候，拒绝接收。

——FC.CTS：继续发送，授权继续发送

——FC.WAIT：请求继续等待

——FC.OVFLW：缓冲溢出，用于指示拆分数据的首帧中字节个数信息超过了接收者可存储的信息总量。

FC.WAIT的值有个最大限值：N\_WFTmax。该参数是系统的常数，且不会在首个 FC N\_PDU中发送。

* 1. 网络层协议数据单元
     1. 协议数据单元类型

不同节点的网络层实体通过交互N\_PDUs实现通讯。

该协议说明了网络层协议数据单元四种类型——单帧（SF N\_PDU），首帧（FF N\_PDU），连续帧（FC N\_PDU）——这些用于建立网络层对等实体通信，交互通信参数，传递通信数据及释放通信资源。

* + 1. SF N\_PDU

SF N\_PDU通过单帧协议控制信息（SF N\_PCI）来标识，SF N\_PDU应当由发送网络实体发送至一个或多个接收网络实体。它应当通过单个的服务请求，传递服务数据及未拆分信息。

* + 1. FF N\_PDU

FF N\_PDU通过首帧协议控制信息（FF N\_PCI）来标识，FF N\_PDU应当由发送网络实体在拆分数据发送期间，发送至特定的接收网络实体。它用于标识拆分数据首帧。接收网络实体在接收到FF N\_PDU时，需重组这些拆分的信息。

* + 1. CF N\_PDU

CF N\_PDU由连续帧协议控制信息标识(CF N\_PCI)。CF N\_PDU传递服务数据<MessageData>的每个段（N\_Data）。发送实体发送所有的N\_PDUs时，FF N\_PDU之后编码成CF N\_PDUs。接收实体在接收到最后一个CF N\_PDU时，重组这些信息并传递至服务的使用者。CF N\_PDU应当由发送网络实体在拆分数据发送期间，发送至特定的接收网络实体。

* + 1. FC N\_PDU

FC N\_PDU由流控协议控制信息(FC N\_PCI)标识。FC N\_PDU指示发送网络实体对CF N\_PDUs的开始，停止或重传。它应当由接收网络层实体在准备好接收更多数据时，发送至发送网络层实体，在接收到如下情况的帧时

1. FF N\_PDU,或者
2. 一组连续帧的最后一个FF N\_PDU，如果另外的一组连续帧需要被发送的话。

FC N\_PDU也能在拆分数据发送期间，通知发送网络层实体暂停CF N\_PDUs的发送或者在检测到首帧中数据（FF\_DL）字长超过接收实体的缓冲区时，中止发送。

* + 1. 协议数据单元域描述
       1. N\_PDU 格式

协议数据单元（N\_PDU）使数据在两个或多个对等网络节点之间传递。所有的N\_PDUs包含了3个域，如下表2所示。

表2 N\_PDU格式

|  |  |  |
| --- | --- | --- |
| 地址信息 | 协议控制信息 | 数据域 |
| N\_AI | N\_PCI | N\_Data |

* + - 1. 地址信息(N\_AI)

N\_AI用于标识对等网络实体间的通信。N\_AI信息在N\_SDU—N\_SA, N\_TA, N\_TAtype, N\_AE—中接收，应当复制包含在P\_PDU中。如果接收到的N\_SDU中<MessageData>及<Length>信息很长，需要网络层拆分这些数据以发送完整的信息，N\_AI应当被复制并包含在每一个要发送的N\_PDU中。

该域包含地址信息标识交互信息类型，数据交互的接收方和发送方。地址信息包含信息地址。

注意：更详细的地址信息的描述，看5.3.2

* + - 1. 协议控制信息（N\_PCI）

该域标识交互的N\_PDUs的类型。它也用来交互在网络层对等实体通信的其它控制参数。

注意： 所有N\_PCI参数更详细的说明见6.5

* + - 1. 数据域（N\_Data）

N\_PDU中的N\_Data用于发送在<MessageData>参数中从服务使用者使用N\_USData.request服务接收的数据。如果必要的话，会在网络发送之前拆分为更小的部分，以适应N\_PDU数据域。

N\_Data的大小依赖N\_PDU的类型及地址格式的选取。

* 1. 协议控制信息说明
     1. N\_PCI

所有的N\_PDU通过N\_PCI来标识，见表3及图4

表3 ——N\_PCI字节概要

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| N\_PDU名 | N\_PCI字节 | | | |
| 字节1 | | 字节2 | 字节3 |
| 7 – 4位 | 3 – 0位 |
| 单帧（SF） | N\_PCItype = 0 | SF\_DL | N/A | N/A |
| 首帧（FF） | N\_PCItype = 1 | FF\_DL | | N/A |
| 连续帧（CF） | N\_PCItype = 2 | SN | N/A | N/A |
| 流控（FC） | N\_PCItype = 3 | FS | BS | STmin |

表4—N\_PCItype值的定义

|  |  |
| --- | --- |
| 16进制值 | 描述 |
| 0 | 单帧  对于未拆分的信息，网络层提供了一个优化的网络协议，即将信息长度值仅放置在PCI字节里。单帧（SF）应当能支持在单个CAN帧中的信息传输。 |
| 1 | 首帧  首帧只支持一条信息无法在单个CAN帧中发送时使用。例如，拆分的信息。拆分信息的第一帧编码为FF，在接收到FF时，接受网络层实体应重组这些信息。 |
| 2 | 连续帧  当发送拆分数据时，所有的连续帧跟着FF编码为连续帧（CF）。在接收到一个连续帧，接受网络层实体应当重组接收到的数据字节直到整个信息被接收到。接收实体在接收最后一帧信息并无接收错误之后，应传递这些信息到相邻的上层。 |
| 3 | 流控帧  流控制的目的是调整CF N\_PDUs发送的速率。流控协议数据单元的3种类型用于支持这些功能。这些类型由协议控制信息的流状态（FS）域指示。 |
| 4 - F | 保留  该范围的值为该协议保留。 |

* + 1. 单帧N\_PCI参数定义

6.5.2.1 SF N\_PCI 字节

表5给出了SF N\_PCI字节总览

表5 SF N\_PCI字节总览

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU名字 | SF N\_PCI byte | | | | | | | |
| Byte 1 | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 单帧 | 0 | 0 | 0 | 0 | SF\_DL | | | |

单帧数据长度（SF\_DL）参数在SF N\_PDU中用于指明服务使用者的字节数。看表6

表6 SF\_DL值的定义

|  |  |
| --- | --- |
| 16进制值 | 说明 |
| 0 | 保留  该范围的值为该协议保留。 |
| 1 – 6 | 单帧数据长度值（SF\_DL）  SF\_DL应编码在N\_PCI字节低位，并分配服务参数<Length>的值。 |
| 7 | 单帧数据长度（SF\_DL）中标准地址  SF\_DL = 7时，只允许标准地址 |
| 8 - F | 无效的  该范围值无效 |

6.5.2.2 SF\_DL出错处理

如果网络层接收到一个SF\_DL=0的单帧（SF），网络层应当忽略接收SF N\_PDU。

如果网络层接收到使用标准地址且一个SF\_DL大于7的单帧，或大于6且使用扩展或混合地址时，网络层应当忽略该SF N\_PDU。

6.5.3 首帧N\_PCI参数定义

6.5.3.1 FF N\_PCI字节

表7 给出了一个FF N\_PCI字节总览

表7 FF N\_PCI字节总览

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU名字 | FF N\_PCI byte | | | | | | | | |
| Byte 1 | | | | | | | | Byte 2 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |
| 首帧 | 0 | 0 | 0 | 1 | FF\_DL | | | | |

6.5.3.2 首帧数据长度（SF\_DL）参数定义

FF N\_PDU中的参数FF\_DL用于说明服务使用者数据字节数。如表8所示，

|  |  |
| --- | --- |
| 16进制数 | 说明 |
| 0 – 6 | 无效的  该范围值无效 |
| 7 | 首帧数据字节（FF\_DL）支持扩展地址及混合地址  FF\_DL = 7 只允许扩展地址及混合地址 |
| 8 - FFF | 首帧数据字节（FF\_DL）  拆分信息在12个位的长度（FF\_DL）上编码，并N\_PCI字节2中最低位置位“0”，N\_PCI字节1中最高位置为“3”。拆分信息最大数据长度支持4095个用户数据。该数据当被分配到服务参数<Length>中。 |

6.5.3.3 FF\_DL 出错处理

如果网络层接收到FF\_DL大于接收方缓冲区的首帧时，应当被认为是错误情况。网络层应当放弃该信息的接收，并且发送包含参数FlowStatus = Overflow的FC N\_PDU。

如果网络层接收到FF\_DL小于8并且使用标准地址，或小于7并且使用扩展地址或混合地址时，网络层应当忽略该首帧并且不必发送一个FC N\_PDU。

6.5.4 连续帧N\_PCI参数定义

6.5.4.1 CF N\_PCI字节

表9 给出了一个CF N\_PCI字节总览

表9 CF N\_PCI字节总览

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU名字 | CF N\_PCI byte | | | | | | | |
| Byte 1 | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 连续帧 | 0 | 0 | 1 | 0 | SN | | | |

6.5.4.2 连续帧参数（SN）定义

CF N\_PDU中参数SN用以说明连续帧的顺序。

——对于所有拆分信息，SN开始于0。FF应当分配值0，它不是明确地包含在N\_PCI域中，但应当按拆分信息顺序号为0。

——第一个流控帧编号（SN）后的连续帧设置为1；

——在同一个拆分信息上，每一个新增的连续帧编号（SN）增1；

——连续帧编号（SN）的值不受流控帧的影响。

——当连续帧编号（SN）到达值15时，它在下一个连续帧中重置为0；

顺序编号如下表10所示

表11所示SN值

表10——SN定义总结

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU | FF | CF | CF | CF | CF | CF | CF | CF |
| SN(hex) | 0 | 1 | … | E | F | 0 | 1 | … |

表11——SN值定义

|  |  |
| --- | --- |
| 16进制值 | 描述 |
| 0 - F | 连续号（SN）  连续号应当在N\_PCI字节1的低字位编码。SN设置值范围在0到15. |

6.5.4.3 SN出错处理

如果接收到一个连续号错误的CF N\_PDU信息，网络层则进行出错处理。信息的接收被终止，并且网络层发送一个<N\_Result>参数=N\_WRONG\_SN的N\_USData.indication指示服务至相邻上层。

6.5.5 流控参数N\_PCI定义

6.5.5.1 流控参数N\_PCI类型

表12 给出了一个FC N\_PCI字节总览

表12 FC N\_PCI字节总览

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU名字 | FC N\_PCI byte | | | | | | | | | |
| Byte 1 | | | | | | | | Byte 2 | Byte 3 |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |  |  |
| 流控帧 | 0 | 0 | 1 | 1 | FS | | | | BS | STmin |

6.5.5.2 流状态参数（FS）定义

流状态参数(FS)指示发送网络实体是否继续信息的发送。

发送网络层实体应当支持所有FS参数规定（不是保留的）的值。

表13——FS值定义

|  |  |
| --- | --- |
| 16进制值 | 说明 |
| 0 | 继续发送（CTS）  流控帧继续发送参数，通过编码N\_PCI第1字节为“0”，表示继续发送。它会促使发送方重新发送连续帧，该值意味着接收者准备好接收最大BS个连续帧。 |
| 1 | 等待（WT）  流控帧等待参数通过编码N\_PCI第1字节为“1”。它会促使发送方继续等待新的流控帧(N\_PDU)的到来，并重新设置N\_BS定时器。 |
| 2 | 溢出（OVFLW）  流控真溢出参数通过编码N\_PCI第1字节为“2”。它会促使发送方中止拆分信息的发送并且做传递参数<N\_Result>=N\_BUFFER\_OVFLW的N\_USData.confirm指示服务。该N\_PCI流控参数值仅能在跟在首帧N\_PDU的流控帧中使用，并且仅能在首帧中FF\_DL信息的长度超过了接收实体缓冲区大小时使用。 |
| 3 - F | 保留  该范围的值为该协议保留 |

6.5.5.3 FS出错处理

如果接收到的FC N\_PDU信息参数出错，网络层进行出错处理。信息的发送被中止，并且网络层传递一个参数<N\_Result>=N\_INVALID\_FS的N\_USData.confirm指示服务至相邻的上层。

6.5.5.4 块大小（BS）参数定义

BS参数应当编码在FC N\_PCI字节2中。

BS单元存储了 每一块中CF N\_PDU的绝对个数。

例如 如果块大小=20（十进制）该块应当包含20个CF N\_PDU。

拆分数据中最后一块连续帧也可能少于BS个帧。

表14提供了FC N\_PCI字节总览

表14 BS值定义

|  |  |
| --- | --- |
| 16进制值 | 说明 |
| 00 | 块大小（BS）  BS参数为0用于指示发送者在拆分数据的发送期间流控制帧不再发送流控制帧了。发送网络层实体应当不停的发送剩下的连续帧以便接收网络层实体另外的流控帧。 |
| 01 - FF | 块大小（BS）  该范围的BS参数值用于指示发送方在没有接收网络实体的流控帧期间能发送的最大数目的连续帧。 |

6.5.5.5 间隔时间(STmin)参数定义

间隔时间（STmin）参数应当编码在FC N\_PCI字节3.

该时间在拆分数据发送过程中，由接收实体指定，并且由发送网络实体遵守。

STmin参数值指定了连续帧协议数据单元发送的最小时间间隔。表15所示。

表15——STmin值定义

|  |  |
| --- | --- |
| 16进制值 | 说明 |
| 00 - 7F | 间隔时间（STmin）范围：0ms - 127ms  该STmin单元的范围00 - 7F 为绝对单位毫秒（ms） |
| 80 - F0 | 保留  该范围值为该协议保留 |
| F1 - F9 | 间隔时间（STmin）范围100us - 900us  该STmin单元的范围F1 - F9最小分编为100微秒（us），参数值F1代表100us，参数值F9代表900us。 |
| FA - FF | 保留  该范围值为该协议保留 |

STmin的度量是在一个连续帧发送完开始到请求下一个连续帧时的间隔时长。

例如 如果STmin=10（十进制），则连续帧网络协议数据单元最小时间间隔=10ms。

6.5.5.6 ST出错处理

在拆分数据发送期间，如果FC N\_PDU信息接收到ST参数值为保留值，发送网络实体则使用最长的ST值，即（7F - 127ms），而不使用从接收网络实体接收到的值。

* 1. FC.Wait帧传递的最大值（N\_WFTmax）

该变量用于避免在通信发送方出现潜在错误挂起的时候，后者可能会持续等待。该参数用于对等通信并不被传递，因此不包含在FC的协议数据单元里。

——N\_WFTmax参数应当指示一组能有多少个FC N\_PDU WT能被接收者接收。

——N\_WFTmax参数的上限由用户根据系统时钟定义。

——N\_WFTmax参数仅由接收网络实体在接收信息的时候使用。

——如果N\_WFTmax参数值设置为0，流控应当继续仅使用FC N\_PDU CTS。流控等待(FC N\_PDU WT)不应再该网络实体中使用。

* 1. 网络层定时
     1. 定时参数

图6显示了网络层定时参数，表16定义了网络层定时参数值及它们相应的给予数据链路服务的开始及结束时间。

运行要求的数值是是对对等通信的约束，以符合该协议。某个应用应指明规定的运行需求，该范围在表16中定义。

定义超时的值应比运行要求的值大保证系统工作且使克服运行需求值在（高总线负载）时，绝对不会满足。指定的超时的值认为是执行的最低限。实际超时的发生不应长于指定超时值+50％。

网络层在检测到错误的时候应传递合适的服务项至服务的使用者。

图6 网络层定制参数置位

表16——网络层定时参数值

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 定时参数 | 描述 | 数据链路服务 | | 超时  (ms) | 运行需求  (ms) |
| Start | End |
| N\_As | 发送方CAN帧发送时间（任何N\_PDU） | L\_Data.request | L\_Data.confirm | 1000 | N/A |
| N\_Ar | 接收方CAN帧发送时间（任何N\_PDU） | L\_Data.request | L\_Data.confirm | 1000 | N/A |
| N\_Bs | 直至下一个流控帧接收的时间 | L\_Data.confirm(FF)  L\_Data.confirm(FC)  L\_Data.indicate(FC) | L\_Data.indicate(FC) | 1000 | N/A |
| N\_Br | 直至下一个流控帧发送的时间 | L\_Data.indicate(FF)  L\_Data.confirm(FC) | L\_Data.request(FC) | N/A | (N\_Br+ N\_Ar)< (0.9\*N\_Bs) |
| N\_Cs | 直到下一个连续帧发送的时间 | L\_Data.confirm(FC)  L\_Data.indication (CF) | L\_Data.request(CF) | N/A | (N\_Cs+ N\_As)< (0.9\*N\_Cr) |
| N\_Cr | 直到下一个连续帧接收的时间 | L\_Data.confirm(FC)  L\_Data.indication (CF) | L\_Data.indication (CF) | 1000 | —— |
| S 发送者  R 接收者 | | | | | |

* + 1. 网络层超时

表17定义了网络层超时的触发和动作

表17 网络层超时出错处理

|  |  |  |
| --- | --- | --- |
| 超时 | 触发 | 动作 |
| N\_As | 发送方没有及时发送N\_PDU | 放弃信息的接收并传递<N\_Result>= N\_TIMEOUT\_A的N\_USData.confirm指示 |
| N\_Ar | 接收方没有及时发送N\_PDU | 放弃信息的接收并传递<N\_Result>= N\_TIMEOUT\_A的N\_USData.confirm指示 |
| N\_Bs | 发送方没有接收到流控帧（丢失，覆盖）或在首帧前收到，或连续帧没有被接收方接收到。 | 放弃信息的发送并传递<N\_Result>= N\_TIMEOUT\_Bs的N\_USData.confirm指示 |
| N\_Cr | 接收方没有收到连续帧或之前流控帧未被发送方收到。 | 放弃信息的接收并传递<N\_Result>= N\_TIMEOUT\_Cr的N\_USData.confirm指示 |

* + 1. 接收到突如其来的N\_PDU

意外的N\_PDU定义为接收到一个节点规则之外的N\_PDU。它可能是该协议定义的某条帧（N\_PDU）（SF N\_PUD, FF N\_PDU, CF N\_PDU 或者 FC N\_PDU），但它接收的却不是按正常的顺序，或者它是一个在本协议中无法解释未知的N\_PDU。

根据网络层支持全双向的或半双向通信的不同，对“意外的”说明也不同：

1. 半双向的，两个节点之间点对点通信在同一个时刻只能是一个方向。
2. 全双向的，两个节点之间点对点通信在同一个时刻支持双向的通信。

除网络层设计决定，使用同一个地址信息（N\_AI）接收或发送到一个节点，认为意外的N\_PDU。

作为一个统一规则，从任何节点过来的意外的N\_PDU应当被忽略，这意味着网络层对该信息的到来无需通知上层。

表18定义了在接收到意外N\_PDU时，网络层动作。考虑到网络层内部状态（NWL状态）及支持半双向或全双向通信。并应知道在N\_PDU接收时候，作为发送者或接收者接收到的N\_PDU包含同一个N\_AI。

表18 ——意外的N\_PDU处理

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| NWL状态 | 接收到。。。 | | | | |
| SF N\_PDU | FF N\_PDU | CF N\_PDU | FC N\_PDU | Unknown  N\_PDU |
| 拆分信息发送过程中 | 全双向通信：  当接受过程中，看该表中相邻下面部分，否则将SF N\_PDU设为新接收的开始 | 全双向通信：  当接受过程中，看该表中相邻下面部分，否则将FF N\_PDU设为新接收的开始 | 全双向通信：  当接受过程中，看该表中相邻下面部分 | 如果等待，则处理该帧，否则忽略 | 忽略 |
|  | 半双向的：  忽略 | 半双向的：  忽略 | 半双向的：  忽略 |  |  |
| 拆分信息接收过程 | 中止当前接收，传递<N\_Result> = N\_UNEXP\_PDU的指示信号至上层，并且将SF N\_PDU为新的接收开始 | 中止当前接收，传递<N\_Result> = N\_UNEXP\_PDU的指示信号至上层，并且将FF N\_PDU为新的接收开始 | 如果等待，则处理该帧，并运行必须的检测（例如SN是否顺序正确）否则忽略 | 全双向的  如果正在发送过程，看相应表上一单元 | 忽略 |
| 半双向的：  忽略 |
| 空闲 | 将SF N\_PDU为新的接收开始 | 将FF N\_PDU为新的接收开始 | 忽略 |  |  |
|  |  |  |  |  |  |

* + 1. 等待帧出错处理

当接收者发送了N\_WFTmax等待流控制帧网络协议数据单元（FC N\_PDU WT）发送方无法通过发送网络协议数据单元（FC N\_PDU CTS）完成流控清除。接收方中止信息的接收，并发送<N\_Result>置位N\_WFT\_OVRN 的N\_USData.indication至相邻上层。

信息的发送者通过接收到<N\_Result>=N\_TIMEOUT\_Bs的N\_USData.confirm确认服务中止信息的接收。（由于丢失了接收方的流控制帧N\_PDU，在发送端产生了一个N\_Bs的超时信号）

* 1. 交错的信息

网络层应当能够实现映射不在一个地址N\_AI的不同信息的传输。这能保证接收者接收网络协议数据单元能持续的重组信息。该策略使网关具有在同时在不同子网处理不同信息传递的功能。

1. 数据链路层的使用

7.1 数据链路层接口服务

7.1.1 L\_Data.request

该请求服务需要通过<Idetifier>标识，将<Data>数据映射到指定的数据链路层单元。

<Idetifier>标识需提供参照指定的访问格式发送<Data>数据：

L\_Data.request(

<Identifier>

<DLC>

<Data>

)

7.1.2 L\_Data.confirm

该确认服务用于确认L\_Data.request请求指定的<Identifier>服务的完成，参数<TransferStatus>提供服务请求的状态。

L\_Data.confirm(

<Identifier>

<TransferStatus>

)

7.1.3 L\_Data.indication

该指示服务指示数据链路层到相邻上层的事件并通过<Identifier>标识发送<Data>数据。

L\_Data.indication (

<Identifier>

<DLC>

<Data>

)

7.2 数据链路层服务参数

下列所示数据链路层服务参数在ISO 11898-1中定义。

<Identifier>: CAN 标识

<DLC>: 数据长度码

<Data>: CAN帧数据

<TransferStatus>: 发送状态

**7.3 映射到N\_PDU域**

**7.3.1 地址格式**

网络层数据交互有三种地址格式的支持：标准，扩展和混合。不同的地址格式需要不同数据长度的CAN帧对包含数据的地址信息进行打包。因此，选择单个CAN帧的数据长度依赖于地址格式类型的选取。

以下（7.3.2到7.3.5）说明了地址格式的映射机制，基于数据链路层服务及11898-1定义的服务参数。

**7.3.2标准地址**

对于N\_SA, N\_TA,N\_TAtype，一个特定的CAN标识符被分配。 N\_PCI和N\_Data安置在CAN帧的数据域。如表19.

表19 —— N\_PDU参数到CAN帧的映射——标准地址

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU类型 | CAN标识 | CAN帧数据域 | | | | | | | |
|  |  | 字节1 | 字节2 | 字节3 | 字节4 | 字节5 | 字节6 | 字节7 | 字节8 |
| 单帧（SF） | N\_AI | N\_PCI | N\_Data | | | | | | |
| 首帧（FF） | N\_AI | N\_PCI | | N\_Data | | | | | |
| 连续帧（CF） | N\_AI | N\_PCI | N\_Data | | | | | | |
| 流控帧（FC） | N\_AI | N\_PCI | | | | N/A | | | |

7.3.3标准混合地址

标准混合地址是标准地址的子格式，也就是映射到CAN标识的地址信息更多一层定义。在上述标准通信，N\_AI和CAN标识之间打开。

对于标准混合通信只允许有29bit的CAN标识。表20和21定义了射到CAN标识的目标地址类型（N\_TAtype）。N\_PCI和N\_Data放在CAN帧数据域。

表20——标准混合地址，N\_TAtype = 物理的

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU类型 | 29bitCAN标识，位地址 | | | | | | | CAN数据域位地址 | | | | | | | |
| 28…26 | 25 | 24 | 23…16 | 15 | 8 | 7…0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 单帧(SF) | 110(bin) | 0 | 0 | 218(dec) | N\_TA | | N\_SA | N\_PCI | N\_Data | | | | | | |
| 首帧(FF) | 110(bin) | 0 | 0 | 218(dec) | N\_TA | | N\_SA |  | | N\_Data | | | | | |
| 连续帧(CF) | 110(bin) | 0 | 0 | 218(dec) | N\_TA | | N\_SA | N\_PCI | N\_Data | | | | | | |
| 流控帧（FC） | 110(bin) | 0 | 0 | 218(dec) | N\_TA | | N\_SA | N\_PCI | | | N/A | | | | |

表21 ——标准混合地址，N\_TAtype = 功能的

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU类型 | 29bitCAN标识，位地址 | | | | | | | CAN数据域位地址 | | | | | | | |
| 28…26 | 25 | 24 | 23…16 | 15 | 8 | 7…0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 单帧(SF) | 110(bin) | 0 | 0 | 219(dec) | N\_TA | | N\_SA | N\_PCI | N\_Data | | | | | | |
| 首帧(FF) | 110(bin) | 0 | 0 | 219(dec) | N\_TA | | N\_SA |  | | N\_Data | | | | | |
| 连续帧(CF) | 110(bin) | 0 | 0 | 219(dec) | N\_TA | | N\_SA | N\_PCI | N\_Data | | | | | | |
| 流控帧（FC） | 110(bin) | 0 | 0 | 219(dec) | N\_TA | | N\_SA | N\_PCI | | | N/A | | | | |

7.3.4 扩展的地址

对于N\_SA, N\_TA,N\_TAtype，一个特定的CAN标识符被分配。N\_TA安置在CAN帧数据域第一个字节，N\_PCI和N\_Data安置在CAN帧数据域剩下的字节。

表22 ——N\_PDU参数到CAN帧的映射——扩展地址

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU类型 | CAN标识 | CAN帧数据域 | | | | | | | |
|  |  | 字节1 | 字节2 | 字节3 | 字节4 | 字节5 | 字节6 | 字节7 | 字节8 |
| 单帧（SF） | N\_AI,无N\_TA | N\_TA | N\_PCI | N\_Data | | | | | |
| 首帧（FF） | N\_AI,无N\_TA | N\_TA | N\_PCI | | N\_Data | | | | |
| 连续帧（CF） | N\_AI,无N\_TA | N\_TA | N\_PCI | N\_Data | | | | | |
| 流控帧（FC） | N\_AI,无N\_TA | N\_TA | N\_PCI | | | N/A | | | |

7.3.5 混合地址

7.3.5.1 29位CAN标识

混合地址是将Mtype设置为远程诊断的地址格式。

表23和24定义了地址信息（N\_AI）到29位CAN标识符的映射机制。主要是目标地址类型（N\_TAtype）,N\_PCI和N\_Data安置在CAN帧数据域剩余字节。

表23——29位CAN标识符的混合地址，N\_TAtype=物理的

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU类型 | 29bitCAN标识，位地址 | | | | | | | CAN数据域位地址 | | | | | | | | |
| 28…26 | 25 | 24 | 23…16 | 15 | 8 | 7…0 | 1 | 2 | 3 | 4 | | 5 | 6 | 7 | 8 |
| 单帧(SF) | 110(bin) | 0 | 0 | 206(dec) | N\_TA | | N\_SA | N\_AE | N\_PCI | N\_Data | | | | | | |
| 首帧(FF) | 110(bin) | 0 | 0 | 206(dec) | N\_TA | | N\_SA | N\_AE | N\_PCI | | | N\_Data | | | | |
| 连续帧(CF) | 110(bin) | 0 | 0 | 206(dec) | N\_TA | | N\_SA | N\_AE | N\_PCI | N\_Data | | | | | | |
| 流控帧（FC） | 110(bin) | 0 | 0 | 206(dec) | N\_TA | | N\_SA | N\_AE | N\_PCI | | N/A | | | | | |

表23——29位CAN标识符的混合地址，N\_TAtype=功能的

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU类型 | 29bitCAN标识，位地址 | | | | | | | CAN数据域位地址 | | | | | | | | |
| 28…26 | 25 | 24 | 23…16 | 15 | 8 | 7…0 | 1 | 2 | 3 | 4 | | 5 | 6 | 7 | 8 |
| 单帧(SF) | 110(bin) | 0 | 0 | 205(dec) | N\_TA | | N\_SA | N\_AE | N\_PCI | N\_Data | | | | | | |
| 首帧(FF) | 110(bin) | 0 | 0 | 205(dec) | N\_TA | | N\_SA | N\_AE | N\_PCI | | | N\_Data | | | | |
| 连续帧(CF) | 110(bin) | 0 | 0 | 205(dec) | N\_TA | | N\_SA | N\_AE | N\_PCI | N\_Data | | | | | | |
| 流控帧（FC） | 110(bin) | 0 | 0 | 205(dec) | N\_TA | | N\_SA | N\_AE | N\_PCI | | N/A | | | | | |

7.3.5.2 11位CAN标识符

混合地址是将Mtype设置为远程诊断的地址格式。

表25定义了地址信息（N\_AI）到11位CAN标识的映射机制。对于N\_SA, N\_TA,N\_TAtype，一个特定的CAN标识符被分配。N\_AE安置在CAN帧数据域的第一个字节。N\_PCI和N\_Data安置在CAN帧数据域的剩余字节。

表25——混合地址（11位CAN标识符）

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| N\_PDU类型 | CAN标识 | CAN帧数据域 | | | | | | | |
| 字节1 | 字节2 | 字节3 | 字节4 | 字节5 | 字节6 | 字节7 | 字节8 |
| 单帧（SF） | N\_AI | N\_AE | N\_PCI | N\_Data | | | | | |
| 首帧（FF） | N\_AI | N\_AE | N\_PCI | | N\_Data | | | | |
| 连续帧（CF） | N\_AI | N\_AE | N\_PCI | N\_Data | | | | | |
| 流控帧（FC） | N\_AI | N\_AE | N\_PCI | | | N/A | | | |

7.4 CAN帧数据长度码（DLC）

7.4.1 DLC参数

DLC参数指定了某一个CAN帧中数据字节长度。本文档不指明数据域中长度的任何要求，而只是在网络层协议数据单元的大小给出暗示。

应用网络层要么将所有CAN帧打包成完整、全部的长度，要么优化DLC以适合网络协议数据单元。

7.4.2 CAN帧数据打包

DLC总是设置为8.如果发送的N\_PDU比8短，那么发送方将DLC设置为最大值为8（打包了一些不使用的数据字节）。这些会在SF，FC帧或拆分信息的最后一个CF帧中会出现。

DLC参数由发送方设置并由接收方决定网络层处理的每一CAN帧的数据字节个数。DLC参数无法用于决定信息长度；该信息应该在信息的开始从N\_PCI的信息中提取。

7.4.3 CAN帧数据优化

DLC不总是为8.如果发送的N\_PDU比8 短，那么发送方会通过缩减CAN帧到只包含N\_PDU占有的字节数（不对无用的数据字节打包）来优化CAN总线负载。CAN帧的优化只能针对SF，FC帧或拆分信息的最后一个CF帧。

DLC参数由发送方设置并由接收方决定网络层处理的每一CAN帧的数据字节个数。DLC参数无法用于决定信息长度；该信息应该在信息的开始从N\_PCI的信息中提取。

7.4.4 数据长度码出错处理

依赖于N\_PCI的值，网络层可计算一个接收到的CAN帧中CAN DLC最小期待的值。

接收到的CAN帧中DLC的值比期待的值要小，（使用打包CAN帧时比8要小或者比网络协议数据单元优化后指示的值要小）网络层应该忽略并不做任何更多的操作。

附件A

（非正式）

按照SAE J1939协议，使用标准与混合地址数据链路层

A.1总览

该附件描述如何映射地址信息参数，按照SAE J1939协议，N\_AI到CAN帧数据链路层

A.2 规则

A.2.1标准地址

表A.1显示了地址信息参数的映射，当网络目标地址类型N\_TAtype使用物理地址，N\_AI到CAN帧的映射。

Table A.1 ——标准地址，物理地址信息

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| J1939协议 | P | R | DP | PF | PS | SA | 数据域 |
| 位 | 3 | 1 | 1 | 8 | 8 | 8 | 64 |
| 内容 | 默认110  (bin) | 0 | 0 | 218(dec) | N\_TA | N\_SA | N\_PCI,  N\_Data |
| CAN标识位 | 28 – 26 | 25 | 24 | 23 – 16 | 15 – 8 | 7 – 0 |  |
| 数据域 |  |  |  |  |  |  | 1 – 8 |
| CAN域 | 标识符 | | | | | | Data |

表A.2显示了地址信息参数的映射，当网络目标地址类型N\_TAtype使用功能地址，N\_AI到CAN帧的映射。

Table A.2 ——标准地址，功能地址信息

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| J1939协议 | P | R | DP | PF | PS | SA | 数据域 |
| 位 | 3 | 1 | 1 | 8 | 8 | 8 | 64 |
| 内容 | 默认110  (bin) | 0 | 0 | 219(dec) | N\_TA | N\_SA | N\_PCI,  N\_Data |
| CAN标识位 | 28 – 26 | 25 | 24 | 23 – 16 | 15 – 8 | 7 – 0 |  |
| 数据域 |  |  |  |  |  |  | 1 – 8 |
| CAN域 | 标识符 | | | | | | Data |

A.2.2混合地址

表A.3显示了地址信息参数的映射，当网络目标地址类型N\_TAtype使用物理地址，N\_AI到CAN帧的映射。

Table A.3 ——混合地址，物理地址信息

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| J1939协议 | P | R | DP | PF | PS | SA | 数据域 | |
| 位 | 3 | 1 | 1 | 8 | 8 | 8 | 8 | 56 |
| 内容 | 默认110  (bin) | 0 | 0 | 206(dec) | N\_TA | N\_SA | N\_AE | N\_PCI,  N\_Data |
| CAN标识位 | 28 – 26 | 25 | 24 | 23 – 16 | 15 – 8 | 7 – 0 |  |  |
| 数据域 |  |  |  |  |  |  | 1 | 2 – 8 |
| CAN域 | 标识符 | | | | | | Data | |

表A.4显示了地址信息参数的映射，当网络目标地址类型N\_TAtype使用功能地址，N\_AI到CAN帧的映射。

Table A.4 ——混合地址，功能地址信息

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| J1939协议 | P | R | DP | PF | PS | SA | 数据域 | |
| 位 | 3 | 1 | 1 | 8 | 8 | 8 | 8 | 56 |
| 内容 | 默认110  (bin) | 0 | 0 | 205(dec) | N\_TA | N\_SA | N\_AE | N\_PCI,  N\_Data |
| CAN标识位 | 28 – 26 | 25 | 24 | 23 – 16 | 15 – 8 | 7 – 0 |  |  |
| 数据域 |  |  |  |  |  |  | 1 | 2 – 8 |
| CAN域 | 标识符 | | | | | | Data | |

A.2.3优先级（P）

优先级由用户自定义，默认为6。

三个位优先级域用于优化CAN总线信息等待时间。优先级域应当接收者屏蔽（忽略）。CAN总线优先级可设置为最高0（000bin），到最低7（111bin）。

A.2.4保留位（R）

应当设置为“0”。

A.2.5数据页（DP）

应当设置为“0”。

A.2.6协议数据单元格式（PF）

格式为PDU1格式，“指明目标地址”

诊断信息应当使用如下参数组（PGN）

——混合地址：52480（dec）即N\_TAtype=功能的，PF=205（dec）。

——混合地址：52736（dec）即N\_TAtype=物理的，PF=206（dec）。

——物理地址：55808（dec）即N\_TAtype=功能的，PF=218（dec）。

——物理地址：56064（dec）即N\_TAtype=物理的，PF=219（dec）。

A.2.7 PDU指定域（PS）

PDU特定域应当包含目标地址，N\_TA

A.2.8源地址（SA）

SA域应当包含源地址，N\_SA

A.2.9更新速率

按照用户的需求

A.2.10 数据长度

数据长度应当为8字节。