



上一篇

UDS快速参考手册

一.网络层协议数据单元

N_PDU

- 1.N_PDU的组成
- 2.N_AI与CAN ID的映射关系
- 3.N_PDU与CAN数据帧的映射关系
- 4.N_PCI协议控制信息
- 5.UDSonCANFD

二.UDS的服务

- 1.UDS的26种服务
 - 2.UDS诊断请求格式
 - 3.UDS诊断响应格式
 - 4.否定响应码NRC
- 三.UDS诊断服务SID详解
- 1.0x10诊断会话控制
 - 2.0x11复位ECU
 - 3.0x27安全访问
 - 4.0x28通信控制
 - 5.0x3E待机握手
 - 6.0x85控制DTC设置
 - 7.0x22通过ID读数据
 - 8.0x2E通过ID写数据
 - 9.0x14清除诊断信息
 - 10.0x19读取DTC信息
 - 11.0x2F通过ID控制IO
 - 12.0x31例行程序控制
 - 13.0x34请求下载
 - 14.0x36数据传输
 - 15.0x37请求退出传输

四.推荐的诊断服务配置方式

回到顶部

UDS快速参考手册

UDS协议是使用最为广泛的车载控制器诊断协议。本参考手册只涉及UDS最基础最常用的部分，更详细的文档请参考ISO标准文件。

一.网络层协议数据单元N_PDU

1.N_PDU的组成

网络层协议数据单元（N_PDU，Network_Protocol Data Unit）包含N_AI，N_PCI，N_Data，即**寻址信息**，**协议控制信息**和**数据**。

- N_AI：Network_Address Information寻址信息，隐含源地址N_SA、目标地址N_TA和寻址方式信息。当使用功能寻址方式通信时，若使用CAN标准帧，则CAN ID为0x7DF。
- N_PCI：Network_Protocol Control Information协议控制信息，标识网络层帧类型。
- N_Data：Network_Data数据，包含应用层协议控制信息和数据。

2.N_AI与CAN ID的映射关系

UDS规定了多种地址格式，其中较常用的是常规寻址与常规固定寻址。常规寻址仅适用于标准帧，在常规寻址方式中，N_AI与CAN ID之间没有确定的映射关系。常规固定寻址仅适用于扩展帧，在常规固定寻址方式中，N_AI与CAN ID之间有确定的映射关系。

对于物理寻址：

CAN ID = 0x18DA[N_TA][N_SA]，其中N_TA占据1个字节，N_SA占据1个字节。

对于功能寻址：

CAN ID = 0x18DB[N_TA][N_SA]，其中N_TA占据1个字节，N_SA占据1个字节。

CAN ID计算示例：诊断仪的物理地址为0xE1，控制器的物理地址为0x0A，则该控制器的

物理寻址ID为0x18DA0AE1。

功能寻址ID为0x18DB0AE1。

3.N_PDU与CAN数据帧的映射关系

N_PDU有四种类型，即单帧（SF）、首帧（FF）、连续帧（CF）、流控帧（FC），用于建立对等实体间的通信。对于不同种类的N_PDU，N_PDU与CAN数据帧的映射关系不同，具体的N_PDU与CAN数据帧的映射关系见下表：

N_PDU类型	CAN标识符	CAN数据帧数据场							
		Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
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				

4.N_PCI协议控制信息

网络层对于四种类型的N_PDU是通过协议控制信息N_PCI进行区分的。每一个N_PDU都只有一个N_PCI。

UDS使用CAN数据场首个半字节(halfbyte)表征N_PCI类型。对于UDS报文，可以通过识别每条CAN报文的首个半字节来确定它属于四种类型中的哪一类。具体的N_PCI协议控制信息见下表：

N_PDU类型	CAN数据帧数据场			
	Byte0		Byte1	Byte2
	Bit7-4	Bit3-0		
单帧(SF)	N_PCIType=0	SF_DL	N_Data	
首帧(FF)	N_PCIType=1	FF_DL		N_Data
连续帧(CF)	N_PCIType=2	SN	N_Data	
流控帧(FC)	N_PCIType=3	FS	BS	STmin

对于单帧（SF），首个字节为0（4bit）+ Data Length(4bit)，控制信息占用1个字节。

1. 对于首帧（FF），前两个字节为1（4bit）+ Data Length(12bit)，控制信息共占用2个字节。
2. 对于连续帧（CF），首字节为2（4bit）+连续帧报文序列号SN，控制信息占用1个字节。序列号SN最大为15，溢出后从0开始重新计数。
3. 对于流控帧（FC），前三个字节为3（4bit）+流状态（FS，4bit）+块大小（BS，8bit）+最小间隔时间（STmin，8bit），控制信息共用三个字节。流状态FS有0：继续发送；1：等待；2：溢出一共3种状态。块大小BS为0时代表数量无限制。最小间隔时间STmin的值小于0x7F时以毫秒为单位（常用），值0xF1 ~ 0xF9代表100us ~ 900us（不常用）。

5.UDSonCANFD

最新的UDS协议不仅支持CAN，还支持CANFD。当报文长度≤8字节时，基于CANFD与基于CAN的UDS网络层协议数据单元N_PDU是完全相同的。当报文长度大于8字节时，基于CANFD与基于CAN的UDS网络层协议数据单元N_PDU是有一些区别的。详细信息可查看下面两张图，图中黄色标记为CANFD特有的内容。

N_PDU类型	CAN标识符	CAN/CANFD数据帧数据场							
		Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
单帧(SF), CAN_DL≤8	N_AI	N_PCI	N_Data						
单帧(SF), CAN_DL>8	N_AI	N_PCI							
首帧(FF), FF_DL≤4095	N_AI	N_PCI	N_Data						
首帧(FF), FF_DL>4095	N_AI	N_PCI							
连续帧(CF)	N_AI	N_PCI	N_Data						
流控帧(FC)	N_AI	N_PCI			N/A				

N_PDU类型	CAN/CANFD数据帧数据场								
	Byte0		Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
	Bit7-4	Bit3-0							
单帧(SF), CAN_DL≤8	N_PCIType=0	SF_DL	N_Data						
单帧(SF), CAN_DL>8	N_PCIType=0	0000	SF_DL	N_Data					
首帧(FF), FF_DL≤4095	N_PCIType=1	FF_DL		N_Data					
首帧(FF), FF_DL>4095	N_PCIType=1	0000	00000000	FF_DL				N_Data	
连续帧(CF)	N_PCIType=2	SN	N_Data						
流控帧(FC)	N_PCIType=3	FS	BS	STmin	填充				

二.UDS的服务

1.UDS的26种服务

UDS的服务分为6大类，一共26种服务，其中15种服务较为常用，具体见下表，表中带背景色的服务为常用服务，后续章节将详细介绍。

大类	SID(0X)	诊断服务名称	Service	子功能支持
诊断和通信管理 功能单元	10	诊断会话控制	Diagnostic Session Control	是
	11	ECU复位	ECU Reset	是
	27	安全访问	Security Access	是
	28	通信控制	Communication Control	是
	3E	待机握手	Tester Present	是
	83	访问时间参数	Access Timing Parameter	是
	84	安全数据传输	Secured Data Transmission	-
	85	控制DTC设置	Control DTC Setting	是
	86	事件响应	Response On Event	是
	87	链路控制	Link Control	是
数据传输功能单元	22	通过ID读数据	Read Data By Identifier	是
	23	通过地址读取内存	Read Memory By Address	-
	24	通过ID读比例数据	Read Scaling Data By Identifier	-
	2A	通过周期ID读数据	Read Data By Periodic Identifier	-
	2C	动态定义数据ID	Dynamically Define Data Identifier	是
	2E	通过ID写数据	Write Data By Identifier	-
	3D	通过地址写内存	Write Memory By Address	-
存储数据传输 功能单元	14	清除诊断信息	Clear Diagnostic Information	-
	19	读取故障码信息	Read DTC Information	是
IO控制功能单元	2F	通过ID控制输入输出	Input Output Control By Identifier	-
例行程序功能单元	31	例行程序控制	Routine Control	是
上传下载功能单元	34	请求下载	Request Download	-
	35	请求上传	Request Upload	-
	36	数据传输	Transfer Data	-
	37	请求退出传输	Request Transfer Exit	-
	38	请求文件传输	Request File Transfer	-

2.UDS诊断请求格式

诊断请求，即诊断方(Tester)给ECU发送指定的请求数据，UDS诊断请求通常有两种类型：

1. SID + Subfunction + Parameter。
2. SID + Parameter。

SID代表需要执行什么诊断服务，长度固定为1个字节。Subfunction代表对这个诊断服务的具体操作，长度也是1个字节。Parameter随诊断服务的不同内容也会发生相应的变化，因此Parameter长度和格式没有统一规定。Parameter的一个重要应用是作为标识符ID，标识诊断请求要读出的数据内容。

3.UDS诊断响应格式

诊断响应，即ECU返回给诊断方(Tester)的响应数据，UDS诊断响应分为两种类型：

- 1. Positive Response肯定响应，意味着诊断仪发过来的诊断请求被正确执行了。肯定响应的格式：Response SID + Subfunction + Parameter或者Response SID + Parameter，其中Response SID 为诊断请求的SID+0x40。
- 2. Negative Response否定响应，意味着当前ECU因为某种原因无法执行诊断仪发过来的诊断请求。否定响应的格式：Negative Response SID + Request SID + Negative Response Code，否定响应长度固定为3个字节，其中Negative Response SID为0x7F，Request SID为诊断请求的SID，Negative Response Code为否定响应的原因。

4.否定响应码NRC

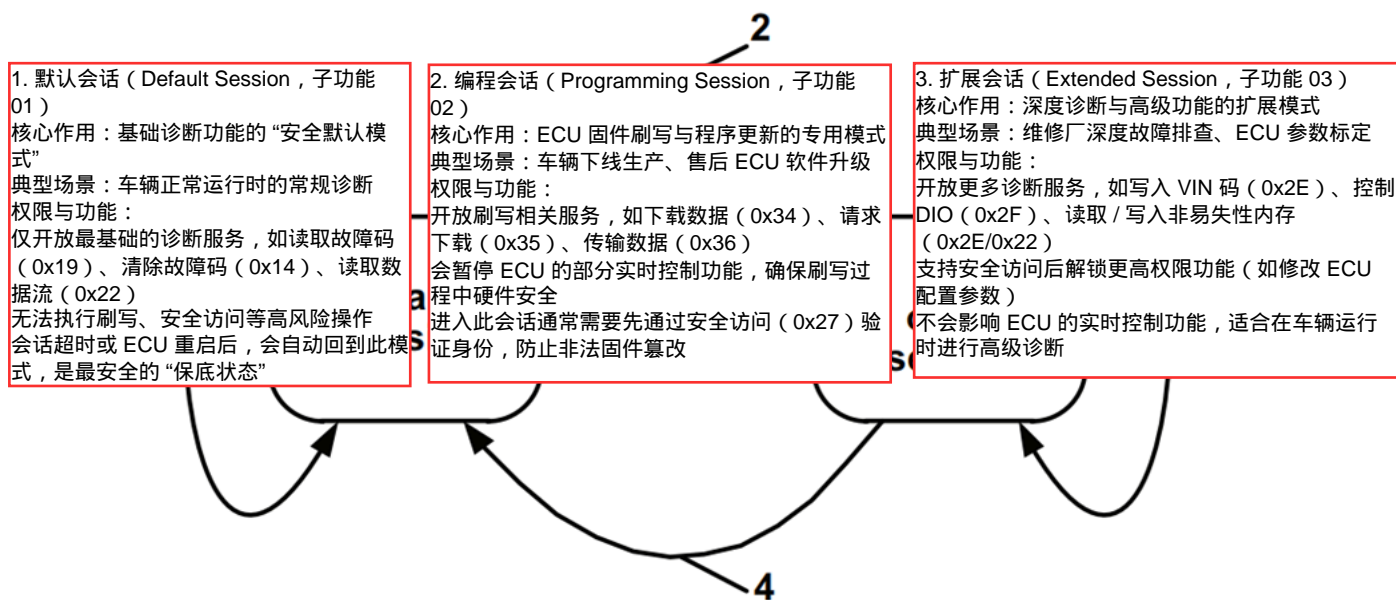
常见的否定响应码NRC请见下表，NRC中0x12、0x13与0x31是最常见的，而有些NRC是一些SID特有的（表格中带括号指定服务的几个NRC）：

NRC	否定响应码	Negative Response Code
0x10	普通拒绝（未定义的服务）	General Reject
0x11	请求的服务不支持	Service Not Supported
0x12	不支持当前请求的子功能	Subfunction Not Supported
0x13	请求报文的长度或者格式不正确	Incorrect Message Length or Invalid Format
0x22	先决条件不正确	Conditions Not Correct
0x24	请求报文的顺序错误	Request Sequence Error
0x31	请求参数超出范围/数据ID不支持	Request Out of Range
0x33	安全访问拒绝，请先解锁	Security Access Denied
0x35	密钥不匹配（0x27服务）	Invalid Key
0x36	尝试解锁次数已达上限（0x27服务）	Exceed Number of Attempts
0x37	超时时间未到（0x27服务）	Required Time Delay Not Expired
0x70	不允许上传/下载（0x34服务）	Upload/Download Not Accepted
0x71	数据传输中止	Data Transfer Suspended
0x72	编程失败（擦除或者烧写失败）	General Programming Failure
0x73	块序列计数错误	Wrong Block Sequence Counter
0x78	请求已接收-即将响应	Request Correctly Received - Response Pending
0x7E	当前会话不支持当前请求的子功能	Subfunction Not Supported in Active Session
0x7F	当前会话不支持请求的服务	Service Not Supported in Active Session
0x92	电压过高	Voltage too High
0x93	电压过低	Voltage too Low

三.UDS诊断服务SID详解

1.0x10诊断会话控制

诊断会话控制服务在ECU中用于切换至不同的诊断会话，对应的SID为0x10。通常包含3个子功能：01 Default（默认会话），02 Programming（编程会话），03 Extended（扩展会话），ECU上电时，进入的是01默认会话。这三个子功能可以相互跳转，当然也可根据实际需求设定这些功能的跳转关系。通常，默认会话模式可以直接切换到扩展会话模式，但是不能直接切换到编程会话模式，如果想进入编程会话模式，则必须先进入扩展会话模式。同样，编程会话模式不能直接进入扩展会话模式，只能进入默认会话模式。



当进入到一个非默认会话时，一个定时器会开始运转，一段时间内没有收到任何请求，一旦时间达到，诊断会退回到默认会话01。当然，可以用3E服务，使诊断一直保持在非默认会话。

诊断会话控制服务CAN报文示例 (使用0x55填充无用字节)：

```
0x02 10 01 55 55 55 55 55 #诊断仪请求ECU跳转到默认会话
0x02 10 02 55 55 55 55 55 #诊断仪请求ECU跳转到编程会话
0x02 10 03 55 55 55 55 55 #诊断仪请求ECU跳转到扩展会话
0x06 50 01 00 32 01 F4 55 #ECU肯定响应，ECU跳转到默认会话
0x03 7F 10 12 55 55 55 55 #ECU否定响应，不支持当前子功能
```

这些响应时间码 (P2Server_max、P2*Server_max) 是 UDS 协议里保障诊断通信稳定性和时效性的核心机制，它们的作用主要有这几点：

1. 明确通信超时边界，避免无限等待

诊断工具 (上位机) 和 ECU 之间的通信必须有明确的超时规则，防止因网络故障、ECU 卡顿导致工具一直等待。

P2Server_max (字节 4 32)：规定了 ECU 对单次请求的最长主响应时间 (这里是 50ms)。如果工具在这个时间内没收到响应，就判定为超时，会触发重发或报错。

P2*Server_max (字节 5 01 + 字节 6 F4)：是分段传输时的辅助响应超时 (这里是 1244ms)。当 ECU 需要分多帧返回大量数据时，每两帧之间的间隔不能超过这个时间，否则工具会认为传输中断。

ECU肯定响应时，需要两个参数P2 (ECU响应最大时间，两个字节，比如上述肯定响应报文中的00 32，单位ms) 与P2* (ECU发送0x78否定响应码之后再次发响应的最大时间，两个字节，比如上述肯定响应报文中的01 F4，单位10ms)。因为0x10是最基础的服务，所有支持UDS的ECU均需要支持，因此通常不会有否定响应。

2.0x11复位ECU

诊断仪通过该诊断服务命令ECU复位。ECU应先发送肯定响应报文，再执行复位，ECU复位成功后，进入默认会话。

ECU复位服务也有几个常用子功能：

- 01 HardReset (硬复位，等同ECU断电，ECU的易失性存储器和非易失性存储器数据可能被重新初始化为默认值)。
- 02 KeyOffOnReset (下电复位，等同车辆熄火，通常会保留非易失性存储器位置的数据，易失性存储器数据被初始化为默认值)。
- 03 SoftReset (软件复位，软件功能回到默认状态，不影响数据)。
- 04 EnableRapidPowerShutDown (快速下电使能，让ECU进入休眠)。

ECU复位服务CAN报文示例：

```
0x02 11 01 55 55 55 55 55 #诊断仪请求ECU硬件复位
0x02 11 03 55 55 55 55 55 #诊断仪请求ECU软件复位
0x02 51 01 55 55 55 55 55 #ECU肯定响应，ECU硬件复位
0x02 51 03 55 55 55 55 55 #ECU肯定响应，ECU软件复位
0x03 7F 11 12 55 55 55 55 #ECU否定响应，不支持当前子功能
```

肯定位 0x51==0x40+0x11(请求SID) + 子功能位
否定位固定为 0x7F + 0x11(请求SID) + 否定响应码(NRC)

ECU肯定响应时，不需要额外的参数。因为0x11是最基础的服务，所有支持UDS的ECU均需要支持，但是并不是每个ECU都支持所有的复位子功能。当ECU不支持某种复位子功能时，ECU有否定响应不支持当前请求的复位子功能。

3.0x27安全访问

车上有很多独有的数据存放在ECU内部，不希望随便被读写；或者一些诊断服务，需要进行安全校验，在Unlocked后才能访问，ECU默认情况下是Locked状态。一般情况下，重启或者重新返回到Default Session都会重新恢复到Locked状态，需要再次Unlocked之后才能进行相应的访问。

完成SecurityAccess安全访问步骤如下：

1. 诊断仪向ECU请求种子（通常使用1、3、5、7子功能）。
2. ECU向诊断仪发送种子。
3. 诊断仪根据种子及与ECU内部相同的算法算出“Key1”，并发送给ECU（通常使用2、4、6、8子功能）。
4. ECU校验诊断仪发来的“Key1”是否有效（ECU也通过相同的Seed和算法计算出Key2，然后比较Key1和Key2的一致性，如果一致则Unlocked）。

安全访问服务CAN报文示例：

```
0x02 27 01 55 55 55 55 55 #诊断仪请求ECU发送种子
0x06 67 01 BB BB BB BB 55 #ECU肯定响应，ECU发送种子
0x06 27 02 CC CC CC CC 55 #诊断仪发送密钥Key1
0x02 67 02 55 55 55 55 55 #ECU肯定响应，ECU解锁
0x03 7F 27 35 55 55 55 55 #ECU否定响应，密钥不匹配
```

否定响应码0x35、0x36与0x37是安全访问服务特有的，其中0x35为基础功能，通常ECU均需要实现，而0x36与0x37则取决于具体ECU的实现情况。

4.0x28通信控制

通信控制服务是控制某类通信关闭/开启接收或者发送。当UDS需要下载升级或者传输大量数据时需要将CAN总线资源让出来，提高传输效率，这时通过0x28服务关闭某类的通信发送报文到CAN总线上，待下载升级或传输数据完成后再通过0x28服务将通信开启即可。

通信控制服务中两个最常用的子功能：00开启通信，03禁止收发。

通信控制服务中的参数为报文类型，通常使用01常规应用报文。

通信控制服务CAN报文示例：

```
0x03 28 00 01 55 55 55 55 #诊断仪请求ECU开启通信
0x03 28 03 01 55 55 55 55 #诊断仪请求ECU禁止通信
0x02 68 00 55 55 55 55 55 #ECU肯定响应，ECU开启通信
0x02 68 03 55 55 55 55 55 #ECU肯定响应，ECU停止通信
0x03 7F 28 12 55 55 55 55 #ECU否定响应，不支持当前子功能
```

ECU否定响应时，有可能是会话模式导致的，因为只有扩展会话模式才支持0x28服务，当ECU处于其它会话模式时，控制器可能发否定响应0x03 7F 28 7E，表示当前会话不支持当前请求的子功能。

5.0x3E待机握手

待机握手服务是告知ECU与诊断仪正连接着，用于保持ECU当前诊断会话（诊断仪一般在ECU处于非Default Session模式下发送0x3E至ECU）。正常情况下，ECU超过一定的时间收不到诊断命令，ECU会退出非Default Session模式，而使用0x3E服务可避免这个问题。

待机握手的两个子功能：00需要响应，80不需要响应。因为子功能80不需要响应通常使用抑制肯定响应来实现，所有ECU往往不需要实现子功能80。

待机握手服务CAN报文示例：

```
0x02 3E 00 55 55 55 55 55 #诊断仪请求ECU保持当前会话，并要求ECU响应
0x02 3E 80 55 55 55 55 55 #诊断仪请求ECU保持当前会话，但不需要ECU响应
0x02 7E 00 55 55 55 55 55 #ECU肯定响应，ECU保持当前会话
0x03 7F 3E 12 55 55 55 55 #ECU否定响应，不支持当前子功能
```

6.0x85控制DTC设置

控制DTC设置服务用于控制ECU的DTC存储，这个服务常常和前面提到的0x28服务一起使用，比如，在开始写参数之前，为了获得更快的传输速度，我们用0x28服务把所有ECU的通信关闭了，但此时因为收到不到相关的报文，ECU会没有必要地存储很多DTC，这时如果使用0x85服务把ECU存储DTC的功能暂时性地禁止掉，则不会造成这种麻烦。

控制DTC设置服务的两个子功能：01开启DTC记录，02关闭DTC记录。

控制DTC设置服务CAN报文示例：


```
0x02 85 01 55 55 55 55 55 #诊断仪请求ECU开启DTC记录
0x02 85 02 55 55 55 55 55 #诊断仪请求ECU关闭DTC记录
0x02 C5 01 55 55 55 55 55 #ECU肯定响应，ECU开启DTC记录
0x02 C5 02 55 55 55 55 55 #ECU肯定响应，ECU关闭DTC记录
0x03 7F 85 12 55 55 55 55 #ECU否定响应，不支持当前子功能
```

ECU否定响应时，有可能是会话模式导致的，因为只有扩展会话模式才支持0x85服务，当ECU处于其它会话模式时，控制器可能发否定响应0x03 7F 85 7E，表示当前会话不支持当前请求的子功能。

7.0x22通过ID读数据

通过ID读数据中的ID指的是Data Identifier数据标识符，简称DID，DID长度为2个字节。诊断仪一次可以请求读取多个DID，ECU将相同个数的DID数据发给诊断仪。但是常用的方式是一次请求读取一个DID数据。

DID有一部分已经被ISO 14229-1规定好了，已经规定好的常用DID见下表：

DID(0x)	Description	说明
F180	bootSoftwareIdentificationDataIdentifier	BOOT软件ID数据ID
F181	applicationSoftwareIdentificationDataIdentifier	应用软件ID数据ID
F182	applicationDataIdentificationDataIdentifier	应用数据ID数据ID
F183	bootSoftwareFingerprintDataIdentifier	BOOT软件指纹数据ID
F184	applicationSoftwareFingerprintDataIdentifier	应用软件指纹数据ID
F185	applicationDataFingerprintDataIdentifier	应用数据指纹数据ID
F186	ActiveDiagnosticSessionDataIdentifier	当前诊断会话数据ID
F187	vehicleManufacturer SparePartNumberDataIdentifier	主机厂车辆备件数据ID
F188	vehicleManufacturer ECUSoftwareNumberDataIdentifier	主机厂车辆ECU软件编号数据ID
F189	vehicleManufacturer ECUSoftwareVersionNumberDataIdentifier	主机厂车辆ECU软件版本编号数据ID
F18A	systemSupplierIdentificationDataIdentifier	系统供应商ID数据ID
F18B	ECUManufacturingDateDataIdentifier	ECU制造日期数据ID
F18C	ECUSerialNumberDataIdentifier	ECU序列号数据ID
F190	VINDataIdentifier	VIN数据ID
F191	vehicleManufacturer ECUHardwareNumberDataIdentifier	主机厂ECU硬件编号数据ID
F192	systemSupplier ECUHardwareNumberDataIdentifier	系统供应商ECU硬件编号数据ID
F193	systemSupplier ECUHardwareVersionNumberDataIdentifier	系统供应商ECU硬件版本编号数据ID
F194	systemSupplier ECUSoftwareNumberDataIdentifier	系统供应商ECU软件编号数据ID
F195	systemSupplier ECUSoftwareVersionNumberDataIdentifier	系统供应商ECU软件版本编号数据ID

通过ID读数据通常不需要使用子功能，使用SID+DID即可发请求。

通过ID读数据服务CAN报文示例：

```
0x03 22 F1 86 55 55 55 55 #诊断仪请求ECU读取当前诊断会话ID
0x04 62 F1 86 01 55 55 55 #ECU肯定响应，ECU发当前诊断会话ID为01
0x03 7F 22 31 55 55 55 55 #ECU否定响应，请求参数超出范围/数据ID不支持
```

8.0x2E通过ID写数据

0x2E的ID与0x22类似，是用来写入数据到ECU。比如车辆下线时，ECU都需要写入VIN号来与车进行绑定。

对于写数据的请求，一般是需要Unlock情况下才能进行，所以在写入DID之前，需要用0x10来进行会话模式转换。

通过ID写数据没有子功能，使用SID+DID即可发请求。

通过ID写数据服务CAN报文示例：

```
0x10 14 2E F1 90 01 02 03 #诊断仪请求ECU写VIN
0x21 04 05 06 07 08 09 0A #连续帧
0x22 0B 0C 0D 0E 0F 10 11 #连续帧
0x03 6E F1 90 55 55 55 55 #ECU肯定响应，VIN已经写入
0x03 7F 2E 31 55 55 55 55 #ECU否定响应，请求参数超出范围/数据ID不支持
```

ECU否定响应时，有可能是访问权限导致的，因为ECU只有在安全状态为解锁状态下才支持0x2E服务，当ECU处于未解锁状态时，控制器可能发否定响应0x03 7F 2E 33，表示安全访问拒绝，需要先解锁ECU。

9.0x14清除诊断信息

客户端通过0x14清除诊断信息服务来请求清除ECU中存储的诊断信息。当ECU完全处理0x14服务后，ECU应发送肯定响应；即使ECU没有存储DTC内容，也应回复肯定响应。

除非另有说明，否则ECU应清除排放与非排放相关的所有DTC。

清除诊断信息没有子功能，使用SID+DTC组编号即可，DTC组编号的长度为3个字节，当DTC组编号为0xFFFFF时为请求清除ECU中存储的所有诊断信息。

清除诊断信息服务CAN报文示例：

```
0x04 14 FF FF FF 55 55 55 #诊断仪请求ECU清除ECU中存储的所有诊断信息
0x01 54 55 55 55 55 55 55 #ECU肯定响应，ECU清除ECU中存储的所有诊断信息
0x03 7F 14 31 55 55 55 55 #ECU否定响应，请求参数超出范围/数据ID不支持
```

10.0x19读取DTC信息 DTC:诊断故障码 (Diagnostic Trouble Code)

如果系统检测到了一个错误，它将存储为DTC。DTC可表现为：电气线路的故障、通信信号的丢失、排放相关的故障或安全相关的故障等。DTC可以表征错误的位置和错误类型。

读取DTC信息服务常用的子服务有：

- 01 (读取符合掩码条件的DTC数量)，后面的参数是DTC状态掩码，若为01表示想读当前故障，若为08表示想读历史故障，若为09表示当前故障和历史故障都想读。
- 02 (读取符合掩码条件的DTC列表及其状态)，后面的参数是DTC状态掩码，解读同上。
- 04 (根据DTC编号读取快照信息)，也叫冻结帧。
- 06 (根据DTC编号读取扩展信息)。
- 0A (读取ECU支持的所有DTC列表及其状态)。

读取DTC信息服务01子功能CAN报文示例：

```
0x03 19 01 09 55 55 55 55 #诊断仪请求ECU读取DTC数量，包括当前故障和历史故障
0x06 59 01 09 01 00 02 55 #ECU肯定响应，ECU符合要求的DTC有两个
0x03 7F 19 12 55 55 55 55 #ECU否定响应，不支持当前子功能
```

ECU肯定响应时，参数一共4个字节，第一个字节为DTC状态掩码，第二个字节参数为0时代表DTC为ISO15031-6格式，为1时代表DTC为ISO14229格式。DTC数量的长度为2个字节。

读取DTC信息服务02子功能CAN报文示例：

```
0x03 19 02 09 55 55 55 55 #诊断仪请求ECU读取DTC列表，包括当前故障和历史故障
0x07 59 02 09 C1 23 45 01 #ECU肯定响应，ECU报告DTC故障码与状态
0x03 7F 19 12 55 55 55 55 #ECU否定响应，不支持当前子功能
```

ECU肯定响应时，参数一共5个字节，第一个字节为DTC状态掩码，后续3个字节为DTC故障码，最后1个字节为故障状态，故障状态由8个位组合起来表示。其中最常用的0x01表征当前故障正在发生，08表征当前故障至少发生过一次，0F表征当前故障至少发生过两次并且故障正在发生，0E表征当前故障至少发生过两次但是当前没有发生。

如果DTC有多个的话，每个DTC占据4个字节（3字节DTC故障码+1字节故障状态），ECU需要使用多帧报文才能将多个DTC发给诊断仪，这时候需要使用首帧与连续帧。

读取DTC信息服务04子功能CAN报文示例：

```
0x06 19 04 C1 23 45 01 55 #诊断仪请求ECU读取故障码C12345的快照信息即冻结帧
0x10 0F 59 04 C1 23 45 08 #ECU肯定响应，ECU报告故障码C12345的快照信息即冻结帧
0x21 01 02 D1 D2 B1 B2 D3 #ECU肯定响应，快照编号01，快照编号的ID个数是2
0x22 D3 B3 55 55 55 55 #ECU肯定响应，DTC ID 是D1D2（数据为B1B2）、D3D4（数据为B3）
0x03 7F 19 12 55 55 55 #ECU否定响应，不支持当前子功能
```

诊断仪请求ECU读取故障码的快照信息即冻结帧时，参数一共4个字节，前3个字节为故障码，最后1个字节为01时代表读取该DTC快照编号为1的快照信息，为FF时代表读取该DTC的所有快照信息，通常都使用01读取DTC的快照信息（读取DTC的所有快照信息过于复杂易出错）。

ECU肯定响应时，参数前4个字节为DTC故障码+0x08（故障状态，可能为其它数据，比如0x01或者0x0F），第5个字节固定为01，第6个字节为快照编号的ID个数，之后的参数为第1个ID+数据，第2个ID+数据等等。ECU需要使用多帧报文才能将DTC快照信息发给诊断仪，这时候需要使用首帧与连续帧。

11.0x2F通过ID控制IO

通过ID控制IO服务可以通过DID(数据标识符)来进行输入信号的替换和控制零部件负载输出。这是一个用在产线上较多的服务。该报文的请求至少由4个字节组成。第一个字节是2F，第二第三字节是DID。第四字节是Input Output Control Parameter（并不算一个子功能，0x2F服务没有子功能），可以看做IO控制类型。

IO控制类型分为4类：

- 00是控制权还给ECU，Return Control To ECU（常用）。
- 01是复位为默认值，Reset to Default（不常用）。
- 02是冻结当前的状态，Freeze Current State（不常用）。
- 03是短暂接管控制权，Short Term Adjustment（常用）。

若控制类型是00-02这三种，请求报文是4个字节，其中00较为常用，01与02不常用。

若控制类型是03，请求报文的第五字节是控制代码，可以是数字量，比如01是开，00是关；也可以是模拟量，比如空调风门的开度。

通过ID控制IO服务CAN报文示例：

```
0x04 2F D1 D2 00 55 55 55 #诊断仪请求ECU控制D1D2代表的IO
0x05 6F D1 D2 00 02 55 55 #ECU肯定响应，ECU控制D1D2代表的IO，当前为2
0x05 2F D1 D2 03 01 55 55 #诊断仪请求接管D1D2代表的IO并开启
0x05 2F D1 D2 03 00 55 55 #诊断仪请求接管D1D2代表的IO并关闭
0x05 6F D1 D2 03 00 55 55 #ECU肯定响应，诊断仪控制D1D2代表的IO，当前关闭
0x03 7F 2F 31 55 55 55 #ECU否定响应，请求参数超出范围/数据ID不支持
```

ECU肯定响应时，参数通常为4个字节，前两个字节为IOID，第3个字节为IO控制类型（常见为00或者03，取决于诊断仪请求的值），第4个字节为IO的状态，状态字节的值与状态之间的对应关系由用户自行定义。需要注意的是，当IO控制类型发生变化时，ECU肯定响应返回的IO状态可能会延迟一个任务周期更新，因此一般需要间隔一段时间（比如100ms）后再次发送相同请求才能获取最新的IO状态。

ECU否定响应时，有可能是会话模式导致的，因为只有扩展会话模式才支持0x2F服务，当控制器处于其它会话模式时，控制器可能发否定响应0x03 7F 2F 7F，表示当前会话不支持当前请求的服务，需要先将会话模式切换到扩展会话模式。

ECU否定响应时，有可能是访问权限导致的，因为ECU只有在安全状态为解锁状态下才支持0x2F服务，当ECU处于未解锁状态时，控制器可能发否定响应0x03 7F 2F 33，表示安全访问拒绝，需要先解锁ECU。

12.0x31例行程序控制

例行程序控制服务可以通过RID（程序标识符）来进行例行程序的控制，通常用于例行程序的开启与关闭，RID与DID类似，长度为2个字节。

例行程序控制服务常用2个子功能：01开启程序，02关闭程序，03请求例行程序控制结果。

例行程序控制服务CAN报文示例：

```
0x04 31 01 FF 01 55 55 55 #诊断仪请求ECU开启FF01代表的程序
0x05 71 01 FF 01 55 55 55 #ECU肯定响应，ECU开启FF01代表的程序
0x03 7F 31 12 55 55 55 #ECU否定响应，不支持当前子功能
0x04 31 03 FF 01 55 55 55 #诊断仪请求FF01代表的例行程序控制结果
0x05 71 03 FF 01 00 55 55 #ECU肯定响应，ECU返回FF01代表的例行程序控制结果
```

ECU肯定响应时，参数通常为3个字节，前两个字节为RID，第3个字节为例行程序的最新状态，状态字节的值与状态之间的对应关系由用户自行定义。UDS允许使用多个字节来描述例行程序的最新状态，此时肯定响应的参数大于3个字节，但是这种方式并不常用。

ECU否定响应时，有可能是会话模式导致的，因为只有扩展会话模式才支持0x31服务，当控制器处于其它会话模式时，控制器可能发否定响应0x03 7F 31 7E，表示当前会话不支持当前请求的子功能，需要先将会话模式切换到扩展会话模式。

ECU否定响应时，有可能是访问权限导致的，因为ECU只有在安全状态为解锁状态下才支持0x31服务，当ECU处于未解锁状态时，控制器可能发否定响应0x03 7F 31 33，表示安全访问拒绝，需要先解锁ECU。

13.0x34请求下载

请求下载服务主要在启动下载传输前启用，作用是提醒ECU准备接收数据，ECU则通过0x74 Response告诉诊断仪自己是否允许传输，以及自己的接受能力是多大。这个服务主要是用来给ECU下载数据的，最常见的应用就是在Bootloader中，程序下载工具会发起下载请求，以完成ECU程序的升级。

请求下载服务没有子功能。

请求下载服务CAN报文示例：

```
0x10 0B 34 00 44 A1 A2 A3 #诊断仪请求ECU下载
0x21 A4 00 00 00 2F 55 55 #下载的地址为A1A2A3A4，下载的大小为2F
0x04 74 20 FF FF 55 55 55 #ECU肯定响应，块最大为FFFF个字节
0x03 7F 34 31 55 55 55 55 #ECU否定响应，请求参数超出范围/数据ID不支持
```

请求下载服务SID后通常使用10个字节的参数，第1个字节通常使用0x00，第2个字节通常使用0x44，后续4个字节代表下载的地址，最后4个字节代表下载的数据块大小。

ECU肯定响应时，参数一共3个字节，第1个字节通常使用0x20，后续两个字节代表数据块大小的最大值。

ECU否定响应时，否定响应码除了0x31之外还可使用0x70（不允许上传/下载）。

ECU否定响应时，有可能是访问权限导致的，因为ECU只有在安全状态为解锁状态下才支持0x34服务，当ECU处于未解锁状态时，控制器可能发否定响应0x03 7F 34 33，表示安全访问拒绝，需要先解锁ECU。

14.0x36数据传输

数据传输服务在0x34服务得到了正确响应之后，诊断仪就要启动数据传输过程了。0x36服务传输的数据块大小上限由上一步ECU返回的74服务中的参数确定。

数据传输服务没有子功能，服务的参数数量由数据块大小决定，第1个字节是数据块的序列编号，编号从1开始，通常情况下，诊断仪需要使用多帧报文才能将数据库发给ECU，这时候需要使用首帧与连续帧。

数据传输服务CAN报文示例：

```
0x11 2B 36 01 D1 D2 D3 D4 #诊断仪请求ECU数据传输第1个数据块
0x21 D5 D6 D7 D8 D9 DA DB #诊断仪传输数据
0x02 76 01 55 55 55 55 55 #ECU肯定响应，第1个数据块传输完毕
0x03 7F 76 31 55 55 55 55 #ECU否定响应，请求参数超出范围/数据ID不支持
```

ECU肯定响应时，参数为1个字节代表数据块的序列编号。

ECU否定响应时，否定响应码除了0x31之外还可使用0x71（数据传输中止）、0x72（编程失败）、0x73（块序列计数错误）。

15.0x37请求退出传输

请求退出服务用于退出上传下载，即诊断仪通过该诊断服务停止与ECU之间的数据传输。如果之前的0x34和0x36服务都顺利执行完成，那么37服务就可以得到ECU的肯定响应。否则ECU会负响应NRC 0x24，表示诊断请求序列有错误。

请求退出服务没有子功能，也没有参数。

请求退出服务CAN报文示例：

```
0x01 37 55 55 55 55 55 55 #诊断仪请求退出传输
0x01 77 55 55 55 55 55 55 #ECU肯定响应，退出传输
0x02 7F 37 24 55 55 55 55 #ECU否定响应，请求序列错误，数据传输失败
```

诊断请求序列有错误通常是因为请求下载的数据大小与实际数据传输的数据大小不一致，对于已经调试完成测试通过的系统，发生错误的概率很小。

四.推荐的诊断服务配置方式

推荐的UDS诊断服务配置方式请见下表：

SID(0X)	Service	默认会话	编程会话	扩展会话	物理寻址	功能寻址
10	Diagnostic Session Control	Y	Y	Y	Y	Y
11	ECU Reset	Y	Y	Y	Y	Y
27	Security Access	-	Y	Y	Y	-
28	Communication Control	-	-	Y	Y	Y
3E	Tester Present	Y	Y	Y	Y	Y
85	Control DTC Setting	-	-	Y	Y	Y
22	Read Data By Identifier	Y	Y	Y	Y	Y
2E	Write Data By Identifier	-	Y*	Y*	Y*	-
14	Clear Diagnostic Information	Y	-	Y	Y	Y
19	Read DTC Information	Y	-	Y	Y	-
2F	Input Output Control By Identifier	-	-	Y*	Y*	-
31	Routine Control	-	-	Y*	Y*	-
34	Request Download	-	Y*	-	Y*	-
36	Transfer Data	-	Y*	-	Y*	-
37	Request Transfer Exit	-	Y*	-	Y*	-

上述配置表格中，Y代表支持，-代表不支持，Y*代表仅在安全状态为解锁状态下才支持，否则不支持。

CANFD.net 提供的内容仅用于培训和测试，不保证内容的正确性。通过使用本站内容随之而来的风险与本站无关。版权所有，保留一切权利。