

Canfd入门

参考如下:

视频参考:

文档参考:

[CAN FD 介绍](#)

学习进度

☐ 一、CAN FD

• []

一、CAN FD

1.原因

随着电动汽车，无人驾驶汽车技术的快速发展，以及对汽车高级驾驶辅助系统和人机交互HMI需求的增加，传统的CAN总线在传输速率和带宽等方面越来越显得力不从心，其主要原因如下：

- 1、通常整车CAN网络负载大大超过推荐值(50%)
- 2、CAN消息中只有大约40-50%的带宽用于实际数据传输
- 3、总线速率通常被限制在1Mbit/s，在实际使用中的速度更低，大多数情况下为500Kbit/s；在J1939网络中使用250Kbit/s
- 4、最大总线速度受响应机制限制，即错误帧，ACK等
- 5、ACK延迟 = 收发器延迟+总线传播延迟

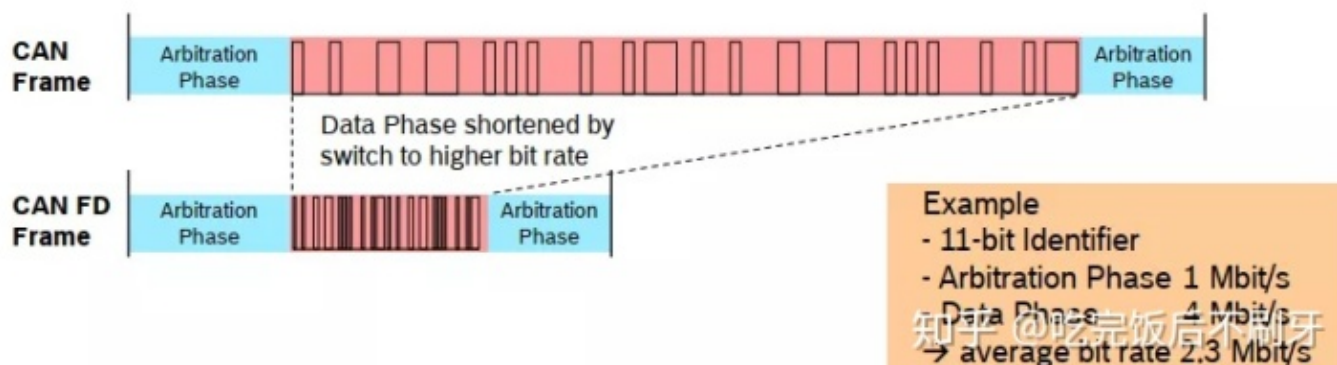
2.CAN FD的出现

为了缩小CAN网络 (Max:1MBit/s)与 FlexRay (Max:10MBit/s)网络的带宽差距，BOSCH推出了CAN FD方案。CAN FD(CAN with Flexible Data rate)继承了CAN总线的主要特性。

CAN总线采用双线串行通讯协议，基于非破坏性仲裁技术，分布式实时控制，可靠的错误处理和检测机制使CAN总线有很高的安全性，但CAN总线带宽和数据场长度却受到制约。CAN FD总线弥补了CAN总线带宽和数据场长度的制约，CAN FD总线与CAN总线的区别主要表现在：

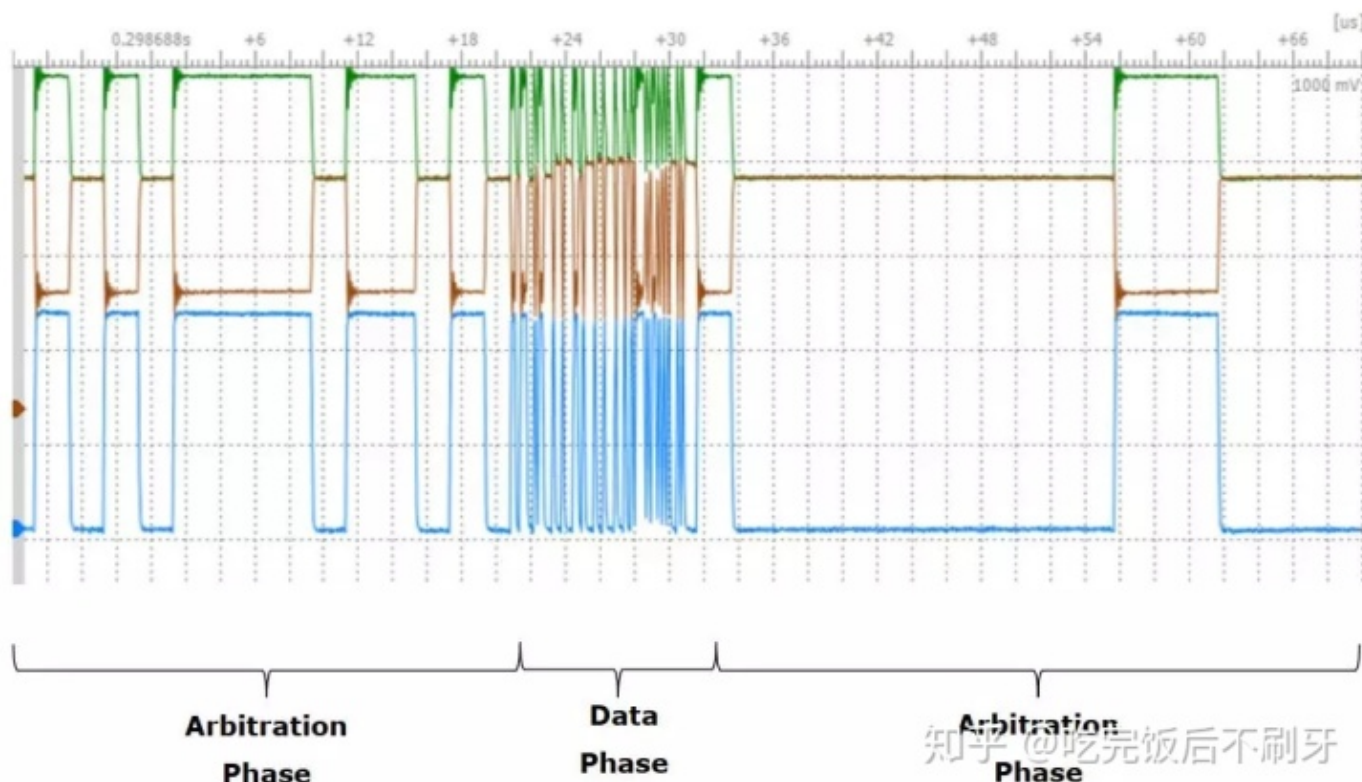
• 可变速率

CAN FD采用了两种位速率：从控制场中的BRS位到ACK场之前（含CRC分界符）为可变速率，其余部分为原CAN总线用的速率。两种速率各有一套位时间定义寄存器，它们除了采用不同的位时间单位TQ外，位时间各段的分配比例也可不同。



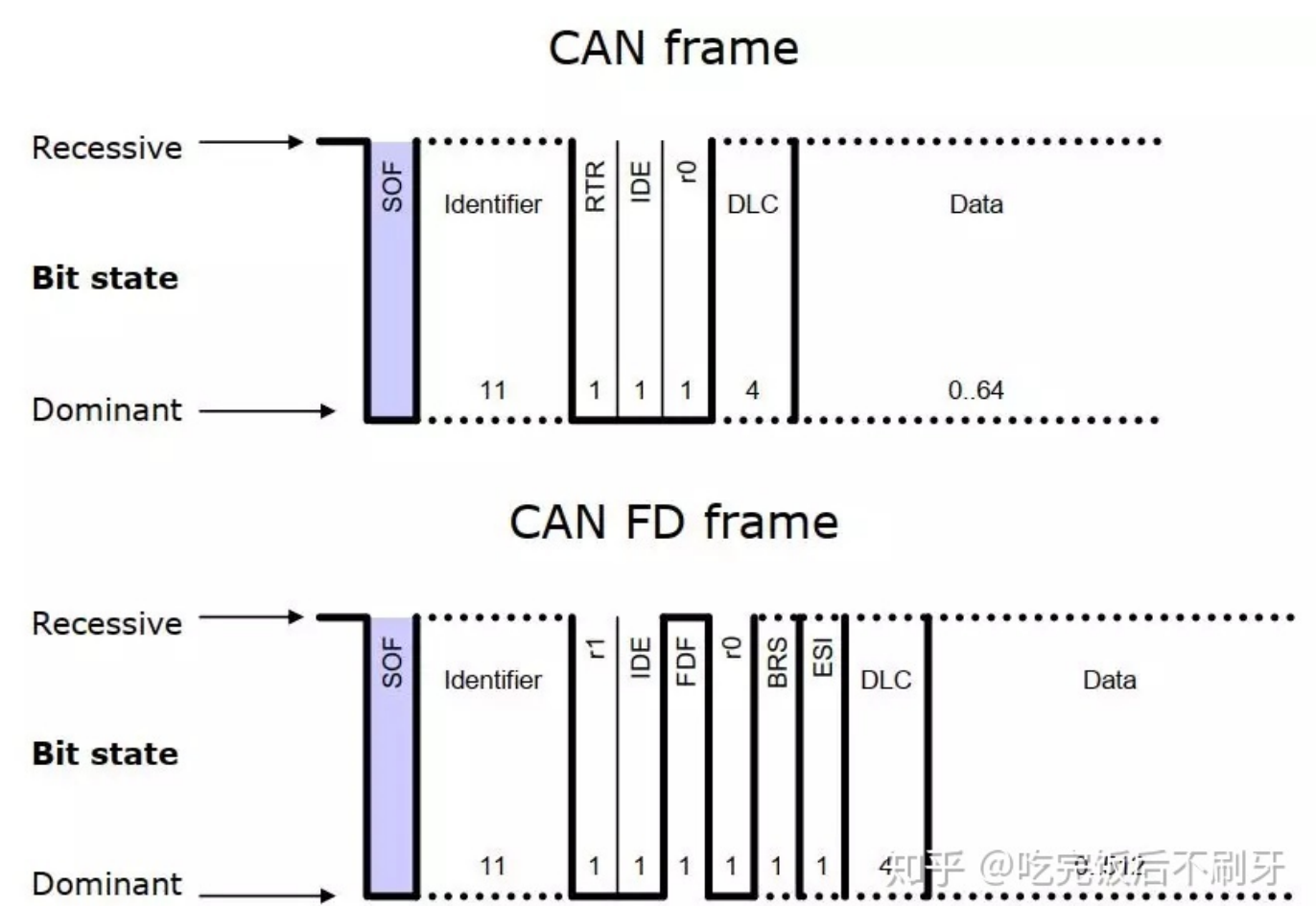
• 新的数据场长度

CAN FD对数据场的长度作了很大的扩充，DLC最大支持64个字节，在DLC小于等于8时与原CAN总线是一样的，大于8时有一个非线性的增长，所以最大的数据场长度可达64字节。



3.CAN FD 结构

与普通CAN报文相同，CAN FD报文一共具有，帧起始SOF，仲裁段，控制段，数据域，CRC域，ACK域，帧结束，共七个部分组成。



- 帧起始

CAN与CANFD使用相同的SOF标志位来标志报文的起始。帧起始由单个显性位构成，标志着报文的开始，并在总线上起着同步作用。

- 仲裁域

与传统CAN相比，CAN FD取消了对远程帧的支持，用RRS位替换了RTR位，为常显性。IDE位仍为标准帧和扩展帧标志位，若标准帧与扩展帧具有相同的前 11 位D，那么标准帧将会由于 IDE 位为 0，优先获得总线。

- RTR(Remote Transmission Request Bit): 远程发送请求位，RTR位在数据帧里必须是显性，而在远程帧里为隐性。
- RRS(Remote Request Substitution): 远程请求替换位，即传统CAN中的RTR位，CAN FD中为常显性。

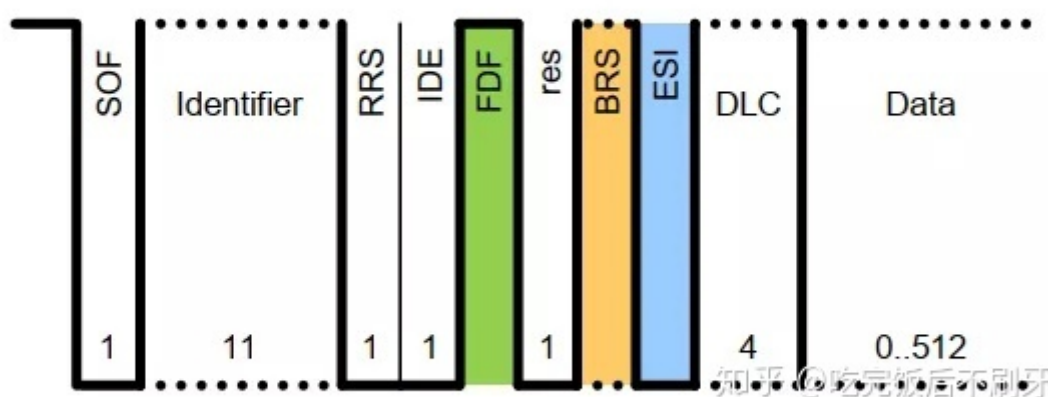
- 控制域

控制域中CANFD与CAN有着相同的IDE, res, DLC位。同时增加了三个控制bit位, FDF、BRS、ESI。

FDF(Flexible Data Rate Format): 原CAN数据帧中的保留位r。FDF常为隐性, 表示CAN FD 报文。

BRS(Bit Rate Switch): 位速率转换开关, 当BRS为显性位时数据段的位速率与仲裁段的位速率一致, 当BRS为隐性位时数据段的位速率高于仲裁段的位速率。

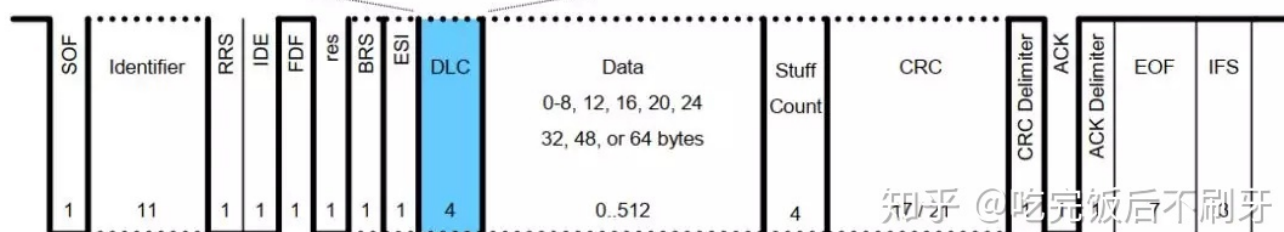
ESI(Error State Indicator): 错误状态指示, 主动错误时发送显性位, 被动错误时发送隐性位。



- DLC

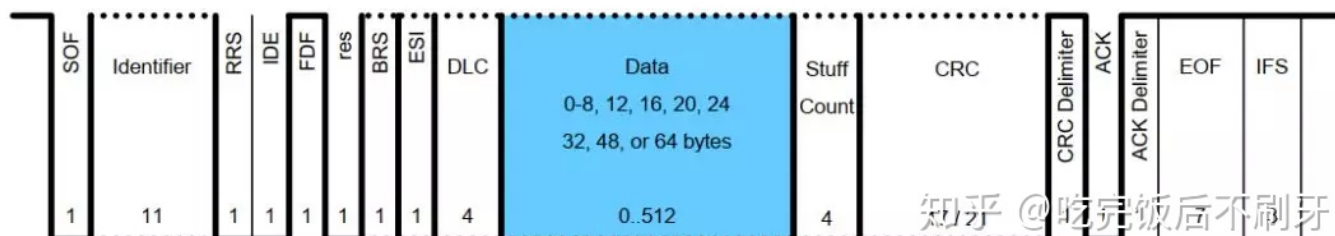
数据域长度位, CAN FD同样使用4bit来确认报文数据场的长度。

1000 = 8	1100 = 24
1001 = 12	1101 = 32
1010 = 16	1110 = 48
1011 = 20	1111 = 64



- 数据域

CAN FD不仅能支持传统的0-8字节报文,同时最大还能支持12, 16, 20, 24, 32, 48, 64字节。



• CRC（循环冗余校验）

该算法是基于一个多项式除法取余的结果,其根据位数需求和多项式变化有数十种版本,凭借其硬件实现简单,位反转纠错能力较强及运算开销适中的优点被广泛应用与数字网络传输以及数据存储领域,如磁盘数据校验、USB、GSM/CDMA通信,在计算机应用中经常接触到的RAR和ZIP文件的压缩/解压数据完整性检查也采用了该算法。

CAN FD对CRC算法作了改变,即CRC以含填充位的位流进行计算。在校验和部分为避免再有连续位超过6个,就确定在第一位以及以后每 4 位添加一个填充位加以分割,这个填充位的值是上一位的反码,作为格式检查,如果填充位不是上一位的反码,就作出错处理。

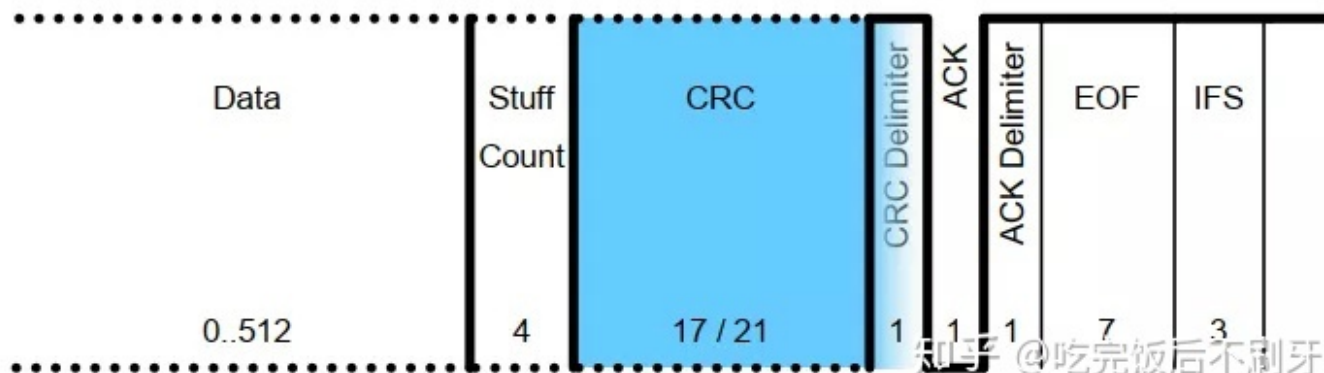
对于数据长度为17-64字节的数据,CAN FD的CRC场扩展到了21位。由于数据场长度有很大变化区间,所以要根据DLC大小应用不同的CRC生成多项式,CRC多项式如下图所示:

Data Length 数据长度	CRC Length CRC 长度	CRC Polynom CRC 多项式
CAN (0-8 字节)	15	$x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + 1$
CAN FD (0-16 字节)	17	$x^{17} + x^{16} + x^{14} + x^{13} + x^{11} + x^6 + x^4 + x^3 + x^1 + 1$
CAN FD (17-64 字节)	21	$x^{21} + x^{20} + x^{13} + x^{11} + x^7 + x^4 + x^3 + 1$

CRC是如何完成校验工作的

CAN帧基于CRC多项式的安全校验,是发送器根据发送的比特计算校验值,并在CAN帧结构CRC字段中提供该结果。接收器使用相同的多项式来计算总线上所见位的校验值,将自我计算的校验值与接收的校准值进行比较,如果匹配,则认为帧被正确接收,接收节点在ACK时隙位中发送显性状态,从而覆盖发送器的隐性状态。在不匹配的情况下,接收节点在ACK定界符之后发送错误帧。

目前CAN FD的控制器CRC校验实现过程会相对复杂一点,在一个CAN总线网络中,帧起始被检测到后所有的节点开始使用三组多项式g15、g17和g21同步计算CRC序列,其中也包含发送节点,由于CRC的计算受CAN帧类型和DLC长度影响,直到CAN帧的控制域以及DLC确认后才选择采用对应的CRC生成序列,确定的CRC序列会在帧结构中CRC字段被采纳用于发送或者用于接收比较。



- ACK & SOF

相比于传统CAN，在CAN FD中最多可接受2个位时间有效的ACK，允许1个额外的位时间来补偿收发器相移和传播延迟）。EOF(End of frame)同样由连续7个隐性位来表示。