

零死角玩转STM32



CAN—通讯实验

淘宝：firestm32.taobao.com

论坛：www.firebbs.com



扫描进入淘宝店铺

01

CAN协议简介

02

STM32的CAN外设简介

03

CAN控制的相关结构体

04

CAN—通讯实验

参考资料:《零死角玩转STM32》

“CAN—通讯实验” 章节

CAN协议简介

CAN是控制器局域网(Controller Area Network)的简称，它是由研发和生产汽车电子产品著称的德国BOSCH公司开发的，并最终成为国际标准（ISO11519），是国际上应用最广泛的现场总线之一。

CAN总线协议已经成为汽车计算机控制系统和嵌入式工业控制局域网的标准总线，并且拥有以CAN为底层协议专为大型货车和重工机械车辆设计的J1939协议。近年来，它具有的高可靠性和良好的错误检测能力受到重视，被广泛应用于汽车计算机控制系统和环境温度恶劣、电磁辐射强及振动大的工业环境。

CAN—通讯实验



CAN物理层

与I2C、SPI等具有时钟信号的同步通讯方式不同，CAN通讯并不是以时钟信号来进行同步的，它是一种异步通讯，只具有CAN_High和CAN_Low两条信号线，共同构成一组差分信号线，以差分信号的形式进行通讯。

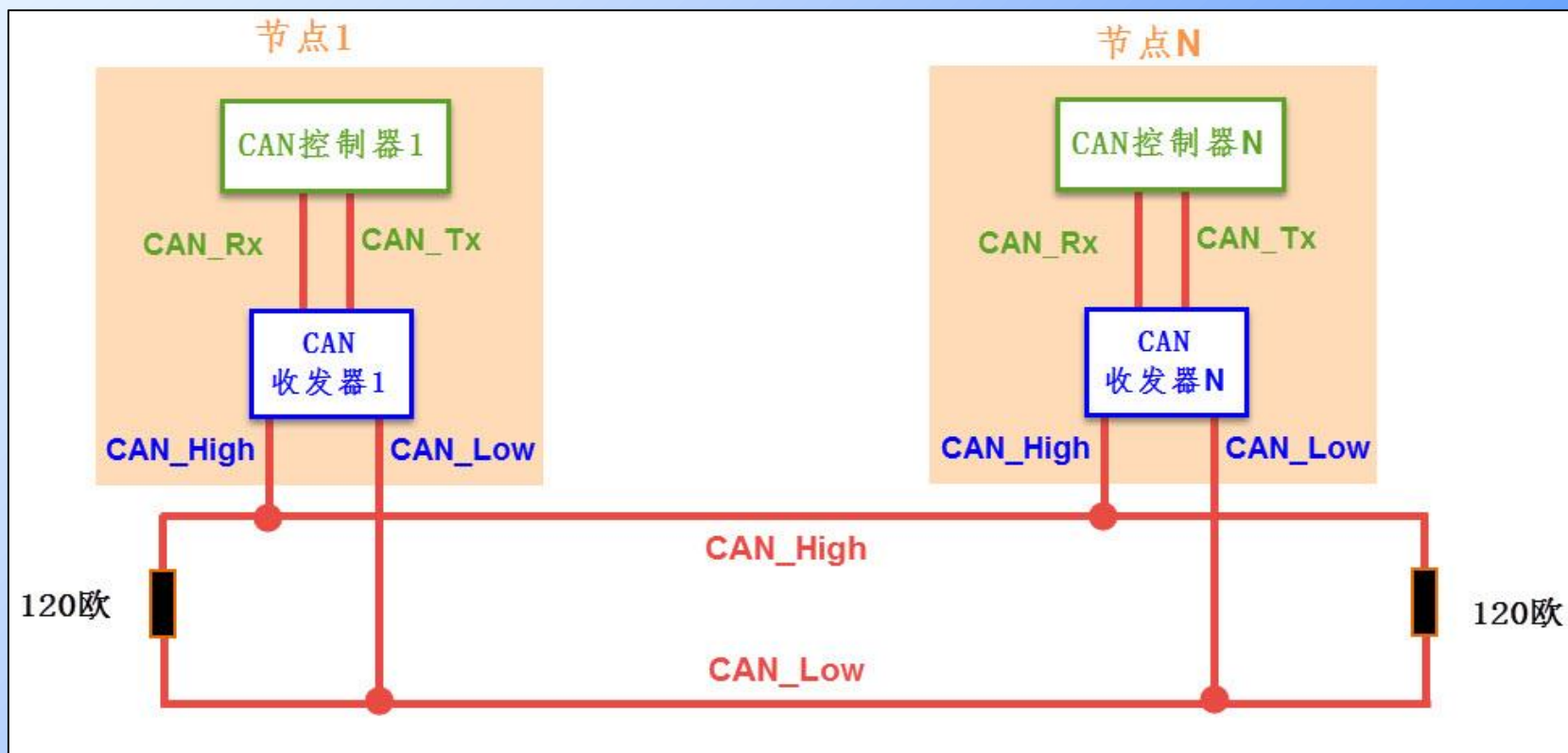
CAN物理层的形式主要分为闭环总线及开环总线网络两种，一个适合于高速通讯，一个适合于远距离通讯。

CAN—通讯实验



闭环总线网络

CAN闭环通讯网络是一种遵循ISO11898标准的高速、短距离网络，它的总线最大长度为40m，通信速度最高为1Mbps，总线的两端各要求有一个“120欧”的电阻。

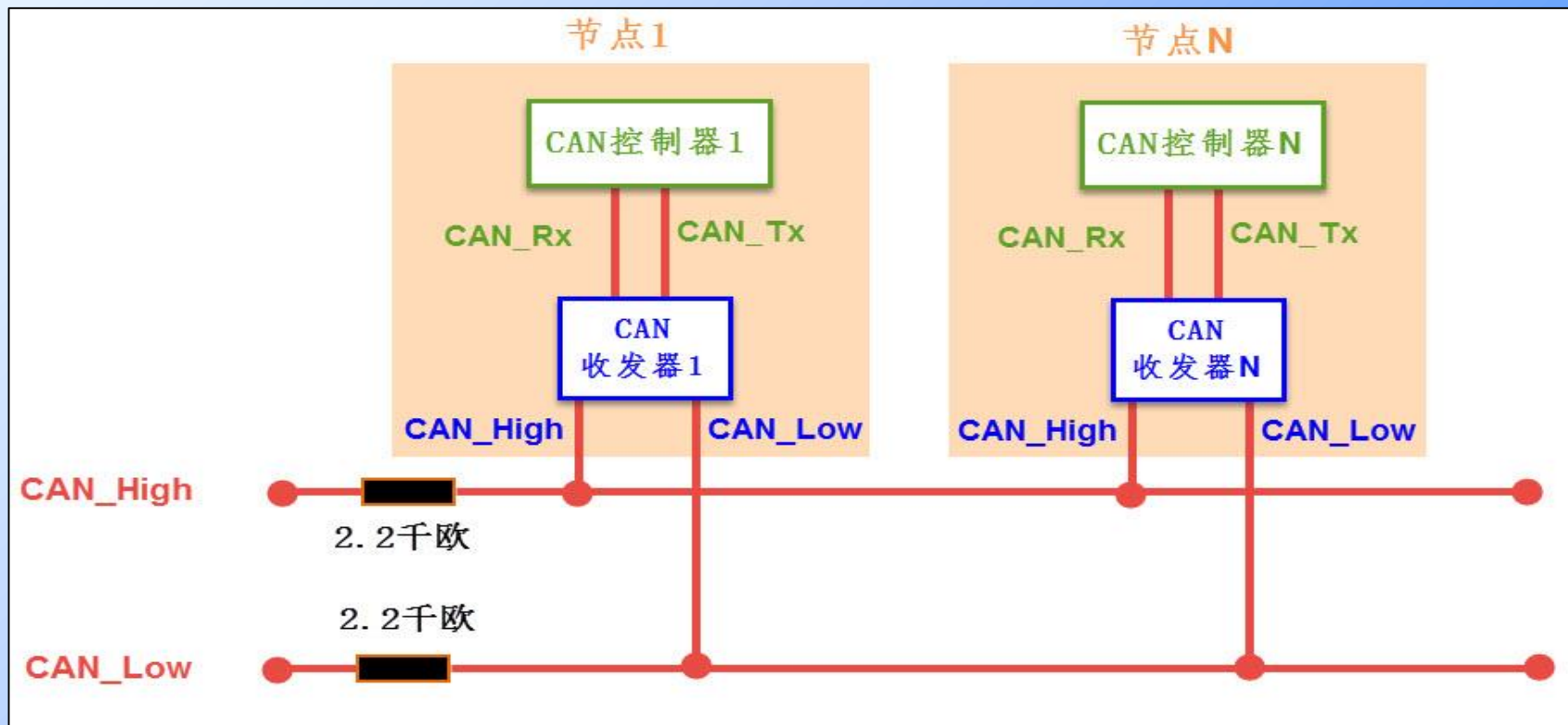


CAN—通讯实验



闭环总线网络

CAN开环总线网络是遵循ISO11519-2标准的低速、远距离网络，它的最大传输距离为1km，最高通讯速率为125kbps，两根总线是独立的、不形成闭环，要求每根总线上各串联有一个“2.2千欧”的电阻。



通讯节点

CAN总线上可以挂载多个通讯节点，节点之间的信号经过总线传输，实现节点间通讯。由于CAN通讯协议不对节点进行地址编码，而是对数据进行编码，所以网络中的节点个数理论上不受限制，只要总线的负载足够即可，可以通过中继器增强负载。

CAN通讯节点由一个CAN控制器及CAN收发器组成，控制器与收发器之间通过CAN_Tx及CAN_Rx信号线相连，收发器与CAN总线之间使用CAN_High及CAN_Low信号线相连。其中CAN_Tx及CAN_Rx使用普通的类似TTL逻辑信号，而CAN_High及CAN_Low是一对差分信号线，使用比较特别的差分信号。

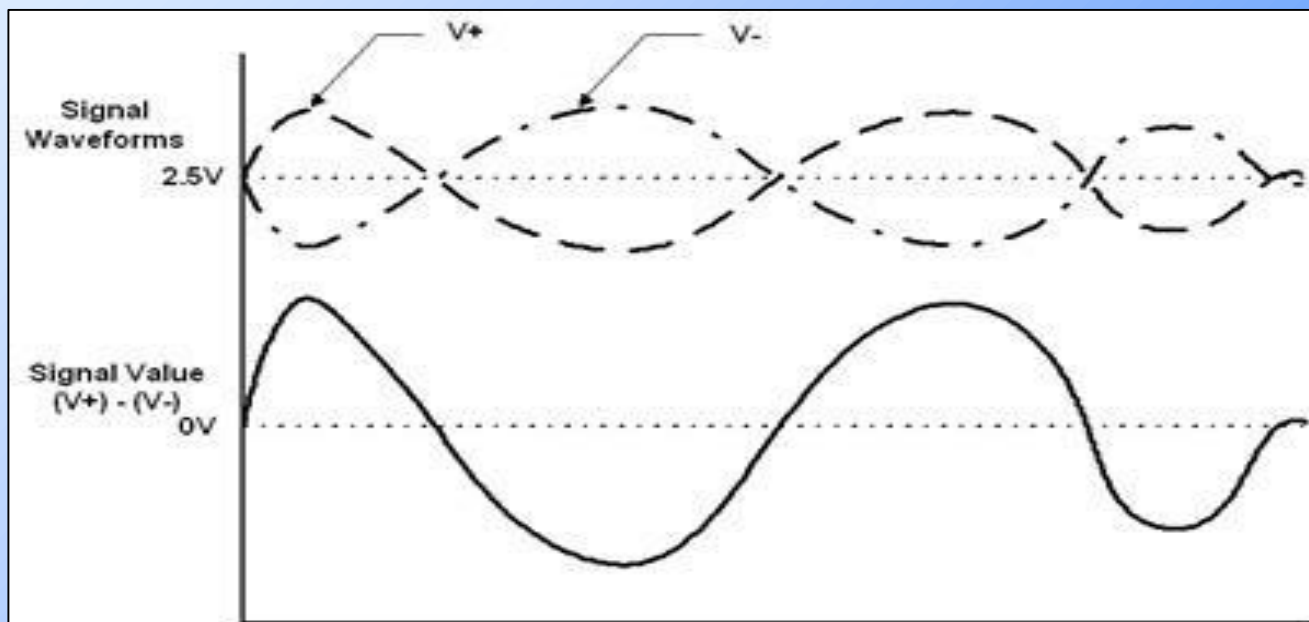
当CAN节点需要发送数据时，控制器把要发送的二进制编码通过CAN_Tx线发送到收发器，然后由收发器把这个普通的逻辑电平信号转化成差分信号，通过差分线CAN_High和CAN_Low线输出到CAN总线网络。而通过收发器接收总线上的数据到控制器时，则是相反的过程，收发器把总线上收到的CAN_High及CAN_Low信号转化成普通的逻辑电平信号，通过CAN_Rx输出到控制器中。

CAN—通讯实验



差分信号

差分信号又称差模信号，与传统使用单根信号线电压表示逻辑的方式有区别，使用差分信号传输时，需要两根信号线，这两个信号线的振幅相等，相位相反，通过两根信号线的电压差值来表示逻辑0和逻辑1。



使用了 V_+ 与 V_- 信号的差值表达出了图下方的信号。

差分信号

相对于单信号线传输的方式，使用差分信号传输具有如下优点：

- 抗干扰能力强，当外界存在噪声干扰时，几乎会同时耦合到两条信号线上，而接收端只关心两个信号的差值，所以外界的共模噪声可以被完全抵消。
- 能有效抑制它对外部的电磁干扰，同样的道理，由于两根信号的极性相反，他们对外辐射的电磁场可以相互抵消，耦合的越紧密，泄放到外界的电磁能量越少。
- 时序定位精确，由于差分信号的开关变化是位于两个信号的交点，而不像普通单端信号依靠高低两个阈值电压判断，因而受工艺，温度的影响小，能降低时序上的误差，同时也更适合于低幅度信号的电路。
- 由于差分信号线具有这些优点，所以在**USB**协议、**485**协议、以太网协议及**CAN**协议的物理层中，都使用了差分信号传输。

CAN—通讯实验



CAN协议中的差分信号

CAN协议中对它使用的CAN_High及CAN_Low表示的差分信号做了规定。

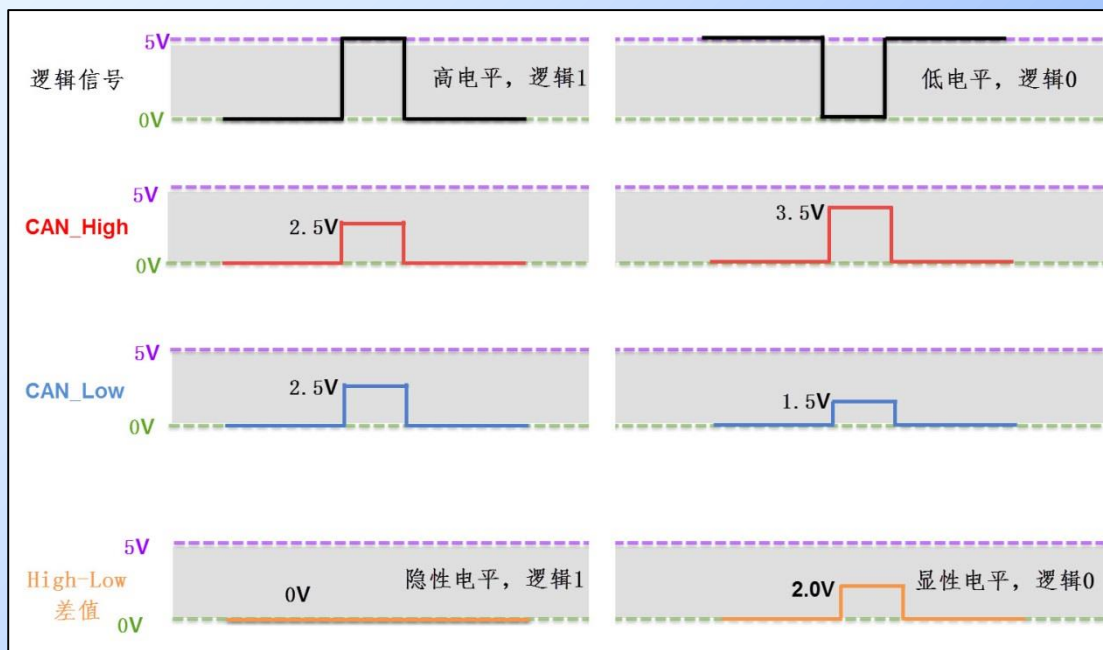
以高速CAN协议为例，当表示逻辑1时(隐性电平)，CAN_High和CAN_Low线上的电压均为2.5v，即它们的电压差 $V_H - V_L = 0V$ ；而表示逻辑0时(显性电平)，CAN_High的电平为3.5V，CAN_Low线的电平为1.5V，即它们的电压差为 $V_H - V_L = 2V$ 。

信号	ISO11898(高速)						ISO11519-2(低速)					
	隐性(逻辑1)			显性(逻辑0)			隐性(逻辑1)			显性(逻辑0)		
	最小值	典型值	最大值	最小值	典型值	最大值	最小值	典型值	最大值	最小值	典型值	最大值
CAN_High (V)	2.0	2.5	3.0	2.75	3.5	4.5	1.6	1.75	1.9	3.85	4.0	5.0
CAN_Low (V)	2.0	2.5	3.0	0.5	1.5	2.25	3.10	3.25	3.4	0	1.0	1.15
High-Low 电位差 (V)	-0.5	0	0.05	1.5	2.0	3.0	-0.3	-1.5	-	0.3	3.0	-

CAN—通讯实验



CAN协议中的差分信号

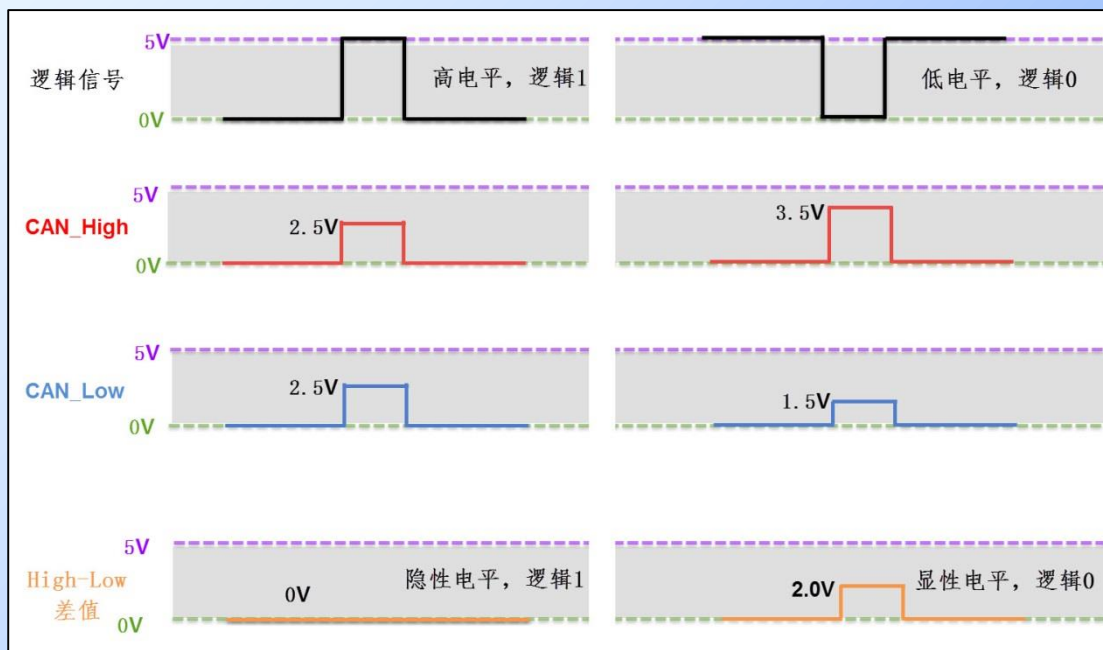


在CAN总线中，必须使它处于隐性电平(逻辑1)或显性电平(逻辑0)中的其中一个状态。假如有两个CAN通讯节点，在同一时间，一个输出隐性电平，另一个输出显性电平，类似I2C总线的“线与”特性将使它处于显性电平状态，显性电平的名字就是这样来的，即可以认为显性具有优先的意味。

CAN—通讯实验



CAN协议中的差分信号



由于CAN总线协议的物理层只有1对差分线，在一个时刻只能表示一个信号，所以对通讯节点来说，CAN通讯是半双工的，收发数据需要分时进行。在CAN的通讯网络中，因为共用总线，在整个网络中同一时刻只能有一个通讯节点发送信号，其余的节点在该时刻都只能接收。。

CAN协议层

CAN的协议层则规定了通讯逻辑。

1. CAN的波特率及位同步

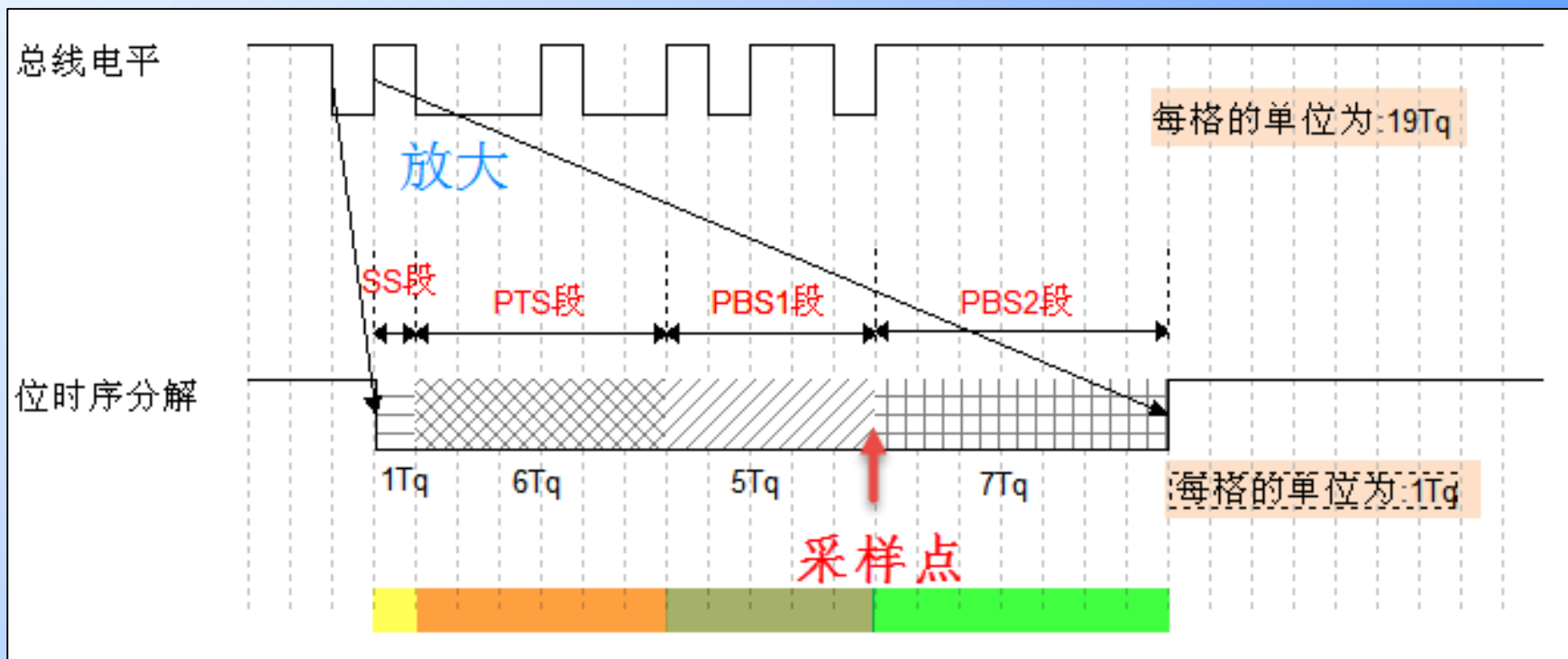
由于**CAN**属于异步通讯，没有时钟信号线，连接在同一个总线网络中的各个节点会像串口异步通讯那样，节点间使用约定好的波特率进行通讯，特别地，**CAN**还会使用“位同步”的方式来抗干扰、吸收误差，实现对总线电平信号进行正确的采样，确保通讯正常。

CAN—通讯实验



位时序分解

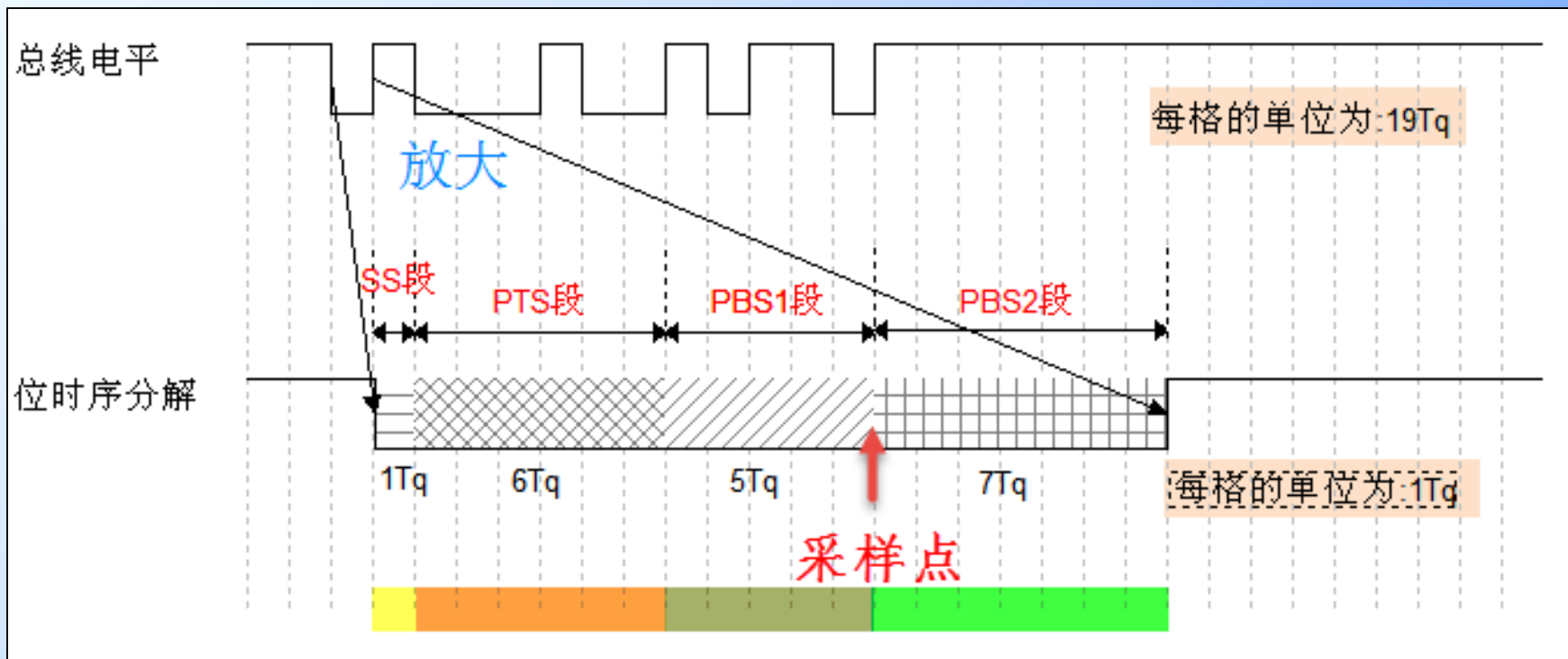
为了实现位同步，CAN协议把每一个数据位的时序分解成SS段、PTS段、PBS1段、PBS2段，这四段的长度加起来即为一个CAN数据位的长度。分解后最小的时间单位是 T_q ，而一个完整的位由8~25个 T_q 组成。



CAN—通讯实验



位时序分解

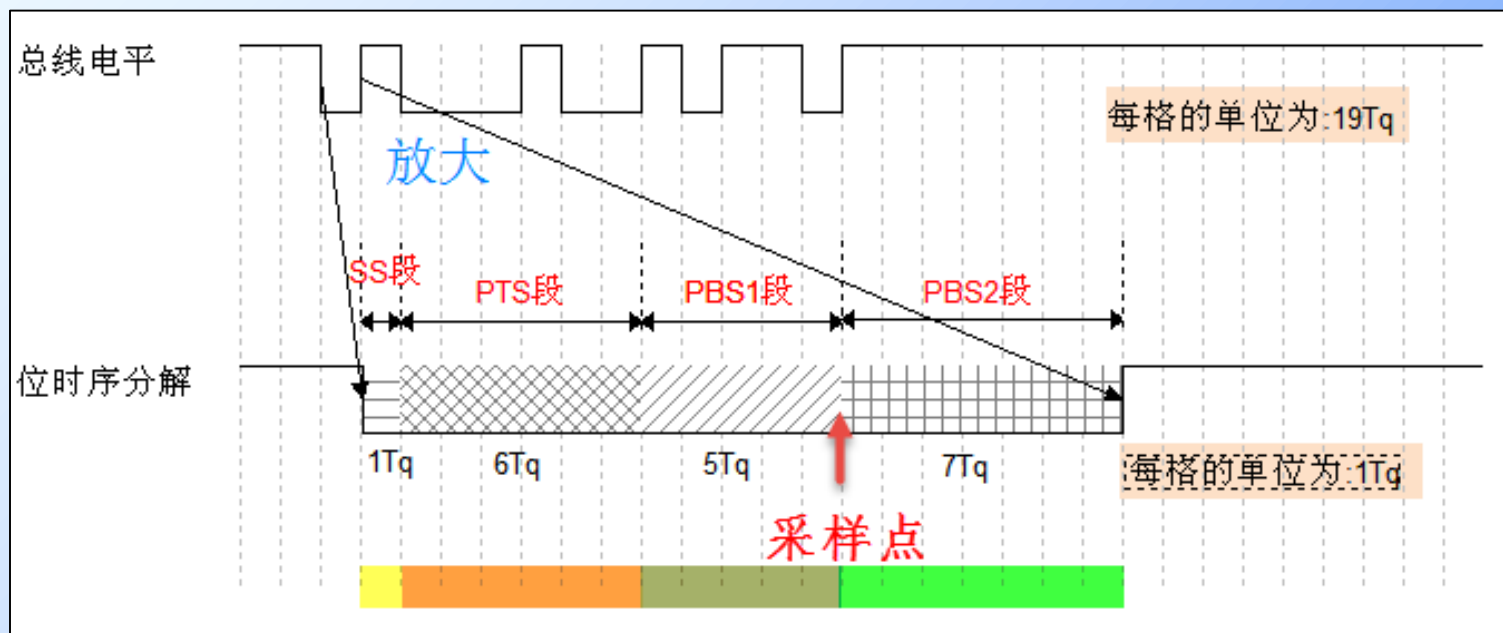


图中表示的CAN通讯信号每一个数据位的长度为 $19Tq$ ，其中SS段占 $1Tq$ ，PTS段占 $6Tq$ ，PBS1段占 $5Tq$ ，PBS2段占 $7Tq$ 。信号的采样点位于PBS1段与PBS2段之间，通过控制各段的长度，可以对采样点的位置进行偏移，以便准确地采样。

CAN—通讯实验



位时序分解



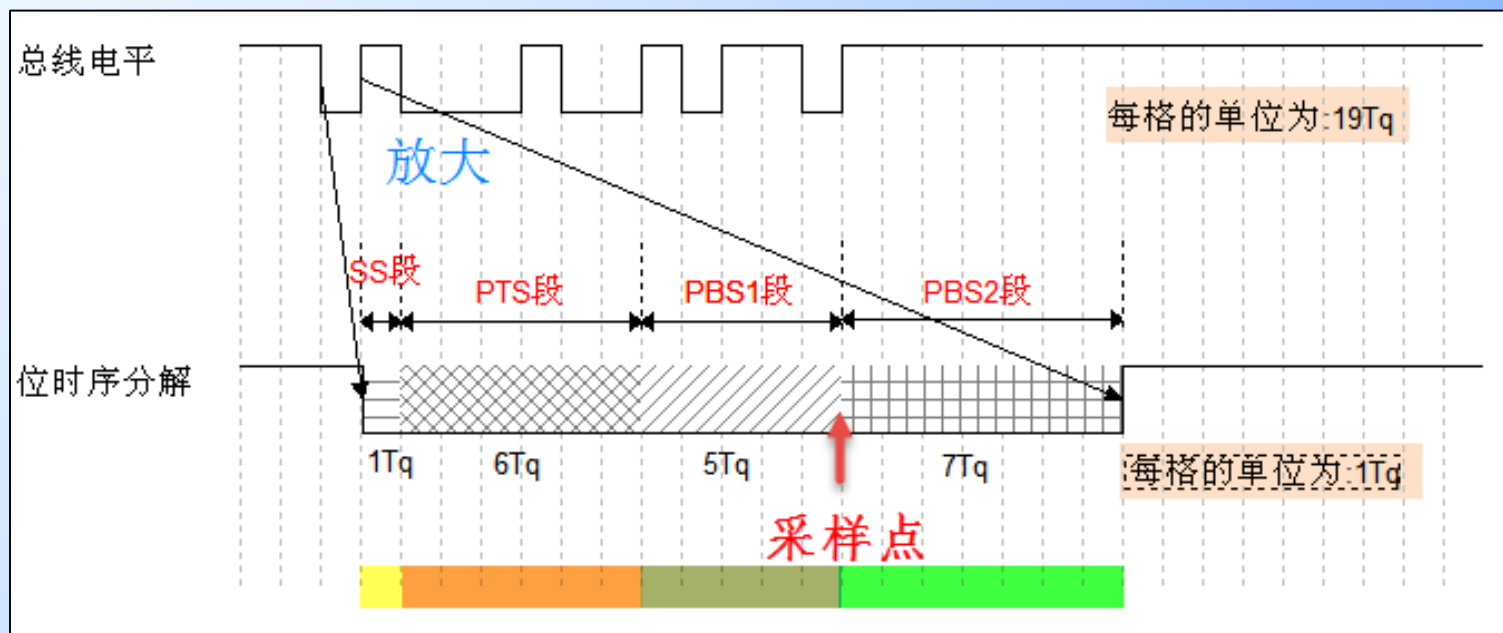
- SS段(SYNC SEG)

SS译为同步段，若通讯节点检测到总线上信号的跳变沿被包含在SS段的范围之内，则表示节点与总线的时序是同步的，当节点与总线同步时，采样点采集到的总线电平即可被确定为该位的电平。SS段的大小固定为 $1Tq$ 。

CAN—通讯实验



位时序分解



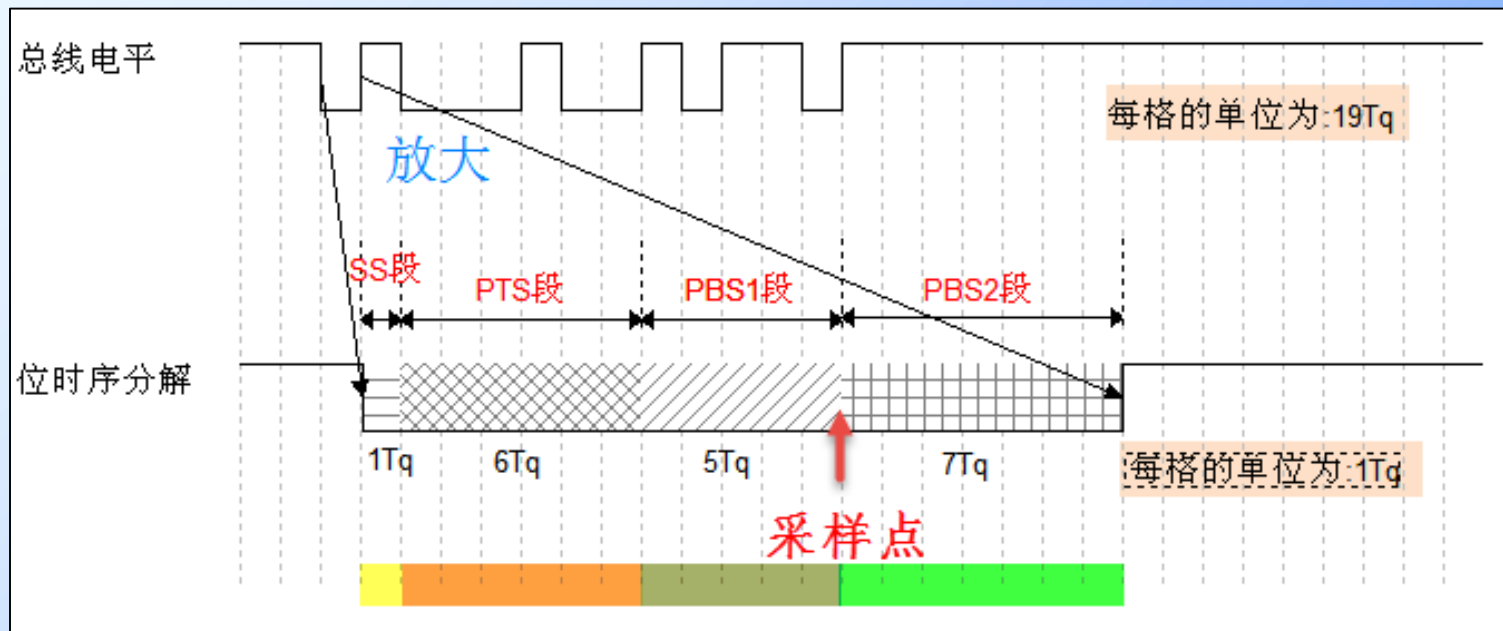
- PTS段(PROP SEG)

PTS译为传播时间段，这个时间段是用于补偿网络的物理延时时间。是总线上输入比较器延时和输出驱动器延时总和的两倍。PTS段的大小可以为1~8Tq。

CAN—通讯实验



位时序分解

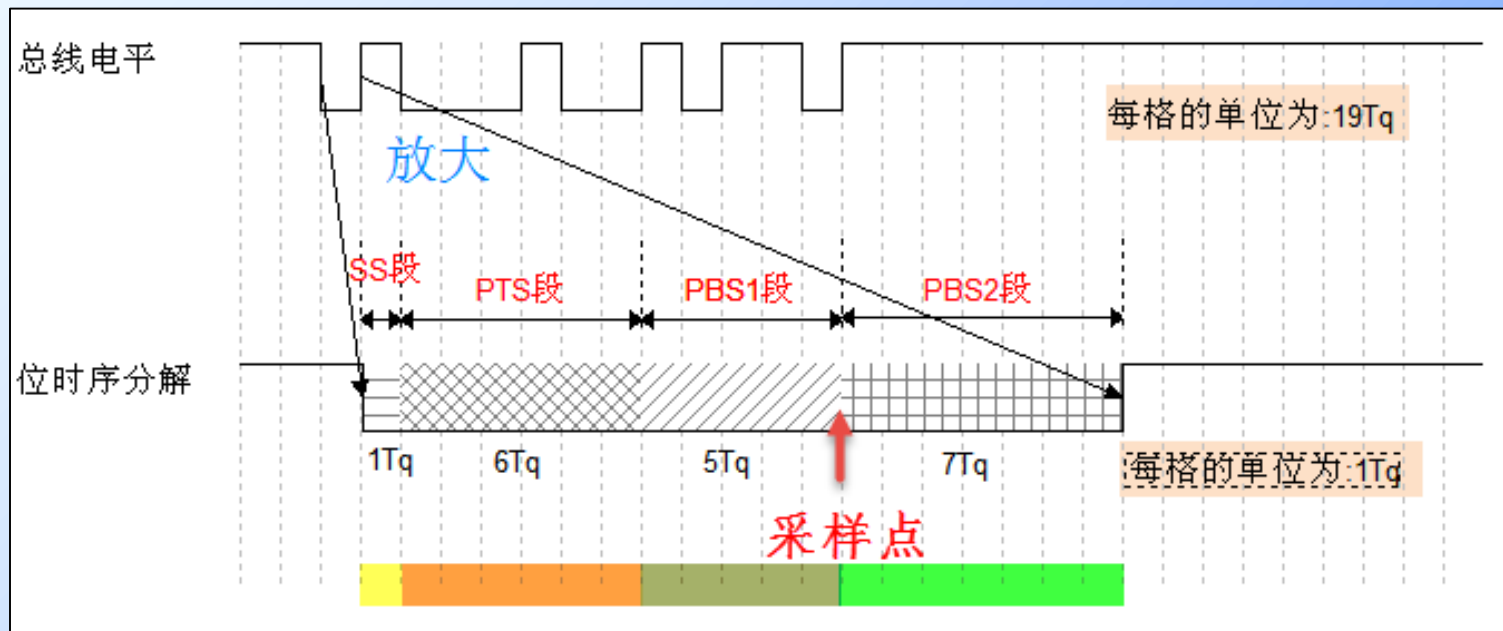


- PBS1段(PHASE SEG1),
PBS1译为相位缓冲段, 主要用来补偿边沿阶段的误差, 它的时间长度在重新同步的时候可以加长。PBS1段的初始大小可以为1~8Tq。

CAN—通讯实验



位时序分解



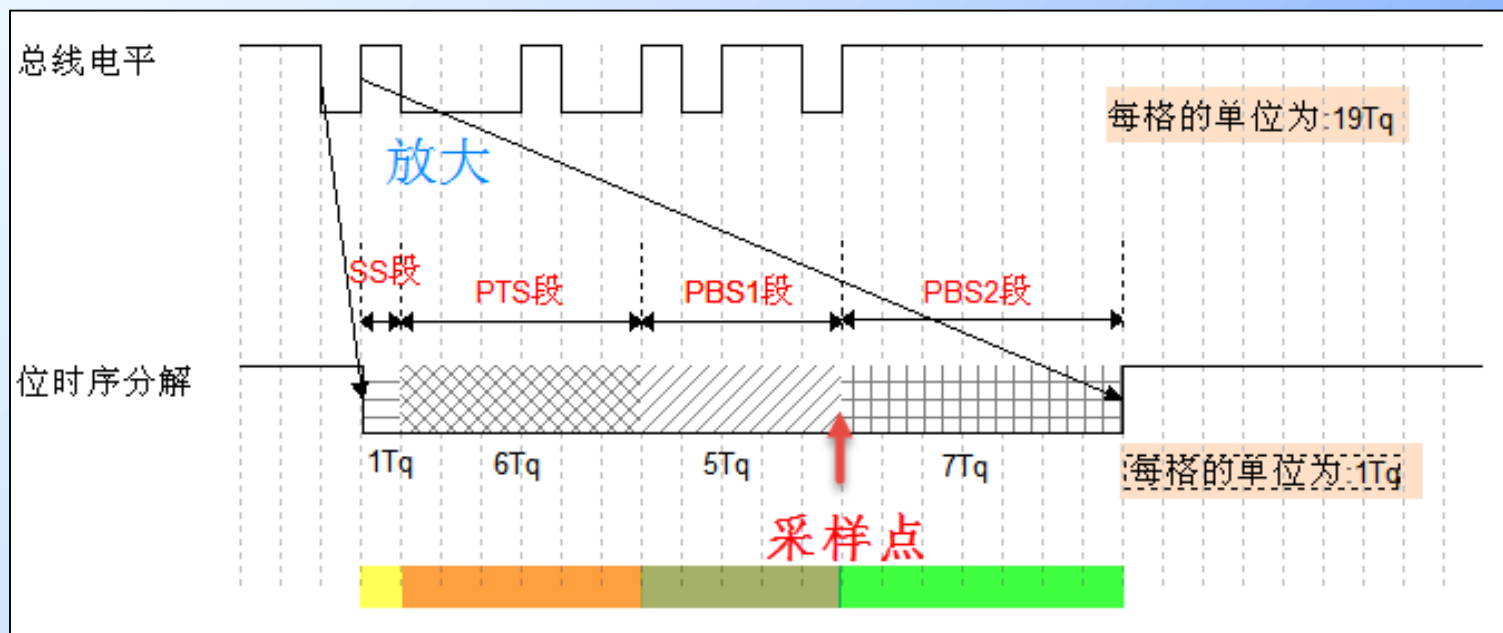
- PBS2段(PHASE SEG2)

PBS2这是另一个相位缓冲段，也是用来补偿边沿阶段误差的，它的时间长度在重新同步时可以缩短。PBS2段的初始大小可以为2~8Tq。

CAN—通讯实验



通讯的波特率



总线上的各个通讯节点只要约定好1个 T_q 的时间长度以及每一个数据位占据多少个 T_q ，就可以确定CAN通讯的波特率。

例如，假设上图中的 $1T_q=1\mu s$ ，而每个数据位由19个 T_q 组成，则传输一位数据需要时间 $T_{1bit}=19\mu s$ ，从而每秒可以传输的数据位个数为：

$$1 \times 10^6 / 19 = 52631.6 \text{ (bps)}$$

这个每秒可传输的数据位的个数即为通讯中的波特率。

2. CAN的报文种类及结构

当使用CAN协议进行通讯时，需要对数据、操作命令(如读/写)以及同步信号进行打包，打包后的这些内容称为报文。

报文的种类

在原始数据段的前面加上传输起始标签、片选(识别)标签和控制标签，在数据的尾段加上CRC校验标签、应答标签和传输结束标签，把这些内容按特定的格式打包好，就可以用一个通道表达各种信号，各种各样的标签就如同SPI中各种通道上的信号，起到了协同传输的作用。当整个数据包被传输到其它设备时，只要这些设备按格式去解读，就能还原出原始数据，这样的报文就被称为CAN的“数据帧”。

CAN—通讯实验



报文的种类

为了更有效地控制通讯，CAN一共规定了5种类型的帧。

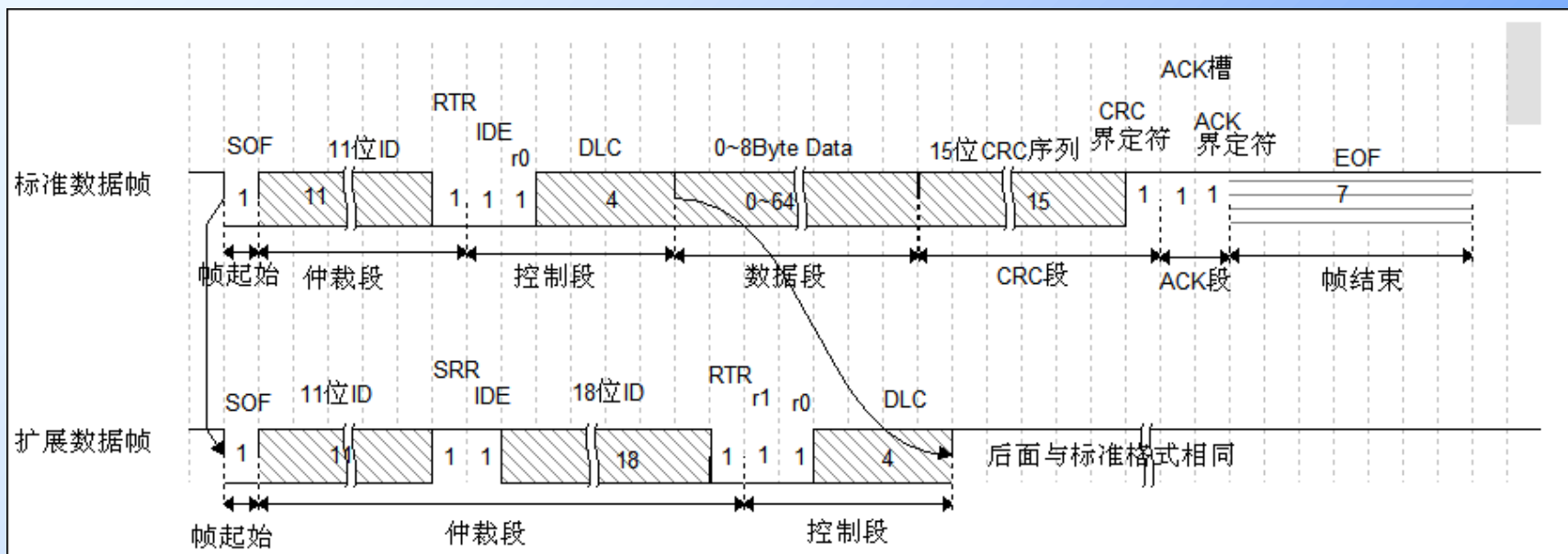
帧	帧用途
数据帧	用于节点向外传送数据
遥控帧	用于向远端节点请求数据
错误帧	用于向远端节点通知校验错误，请求重新发送上一个数据
过载帧	用于通知远端节点：本节点尚未做好接收准备
帧间隔	用于将数据帧及遥控帧与前面的帧分离开来

CAN—通讯实验



数据帧的结构

数据帧的结构图：

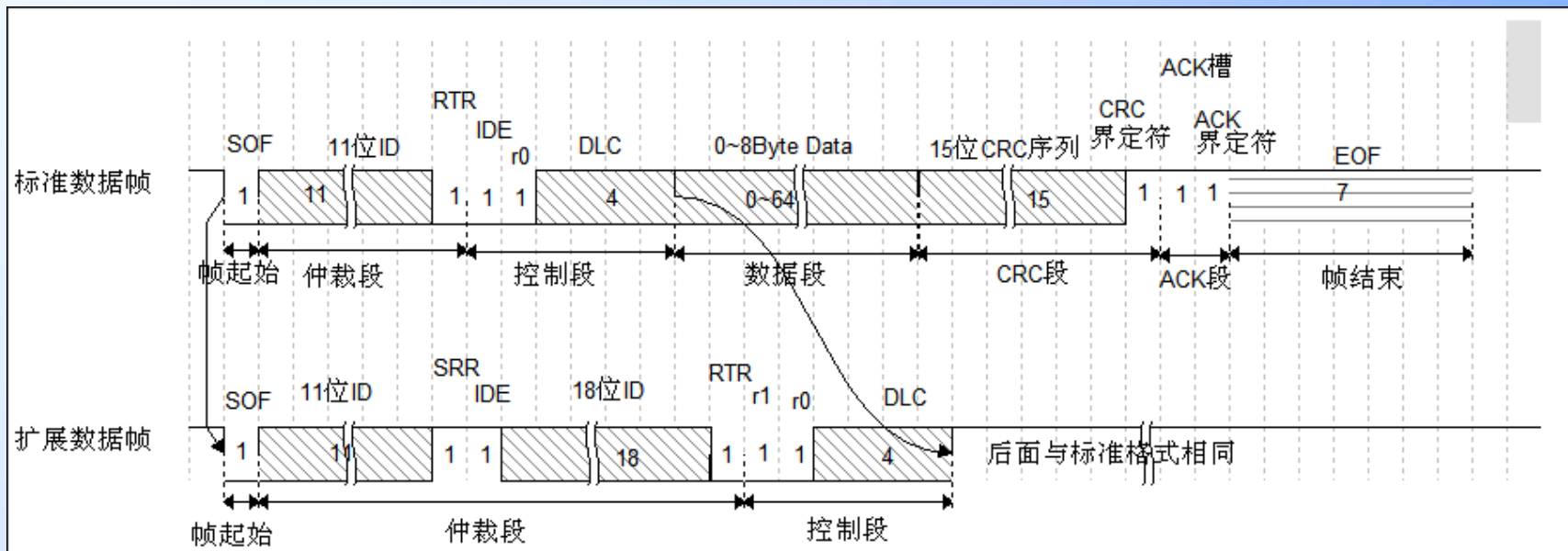


数据帧以一个显性位(逻辑0)开始，以7个连续的隐性位(逻辑1)结束，在它们之间，分别有仲裁段、控制段、数据段、CRC段和ACK段。

CAN—通讯实验



帧起始



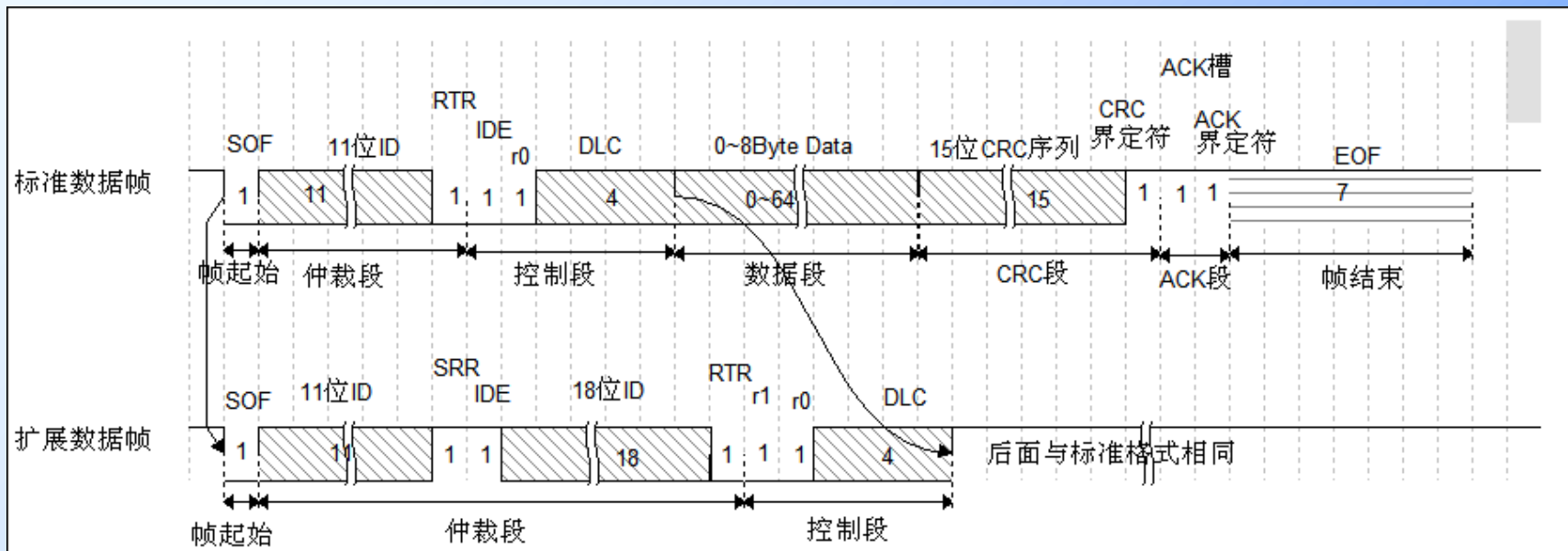
- 帧起始

SOF段(Start Of Frame), 译为帧起始, 帧起始信号只有一个数据位, 是一个显性电平, 它用于通知各个节点将有数据传输, 其它节点通过帧起始信号的电平跳变沿来进行硬同步。

CAN—通讯实验



仲裁段



- 仲裁段

当同时有两个报文被发送时，总线会根据仲裁段的内容决定哪个数据包能被传输，这也是它名称的由来。

仲裁段

仲裁段的内容主要为本数据帧的ID信息(标识符)，数据帧具有标准格式和扩展格式两种，区别就在于ID信息的长度，标准格式的ID为11位，扩展格式的ID为29位，它在标准ID的基础上多出18位。

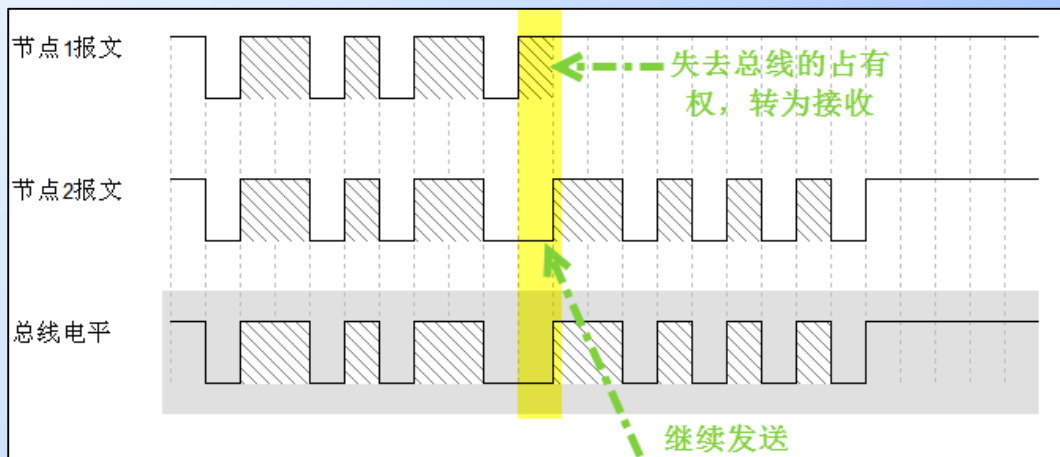
在CAN协议中，ID起着重要的作用，它决定着数据帧发送的优先级，也决定着其它节点是否会接收这个数据帧。CAN协议不对挂载在它之上的节点分配优先级和地址，对总线的占有权是由信息的重要性决定的，即对于重要的信息，可给它打包上一个优先级高的ID，使它能够及时地发送出去。

也正因为它这样的优先级分配原则，使得CAN的扩展性大大加强，在总线上增加或减少节点并不影响其它设备。

CAN—通讯实验



仲裁段



报文的优先级，是通过**对ID的仲裁**来确定的。根据前面对物理层的分析我们知道如果总线上同时出现显性电平和隐性电平，总线的状态会被置为显性电平，**CAN**正是利用这个特性进行仲裁。

若两个节点同时竞争**CAN**总线的占有权，当它们发送报文时，若首先出现隐性电平，则会失去对总线的占有权，进入接收状态。在开始阶段，两个设备发送的电平一样，所以它们一直继续发送数据。到了图中箭头所指的时序处，节点单元**1**发送的为隐性电平，而此时节点单元**2**发送的为显性电平，由于总线的“线与”特性使它表达出显示电平，因此单元**2**竞争总线成功，这个报文得以被继续发送出去。

CAN—通讯实验



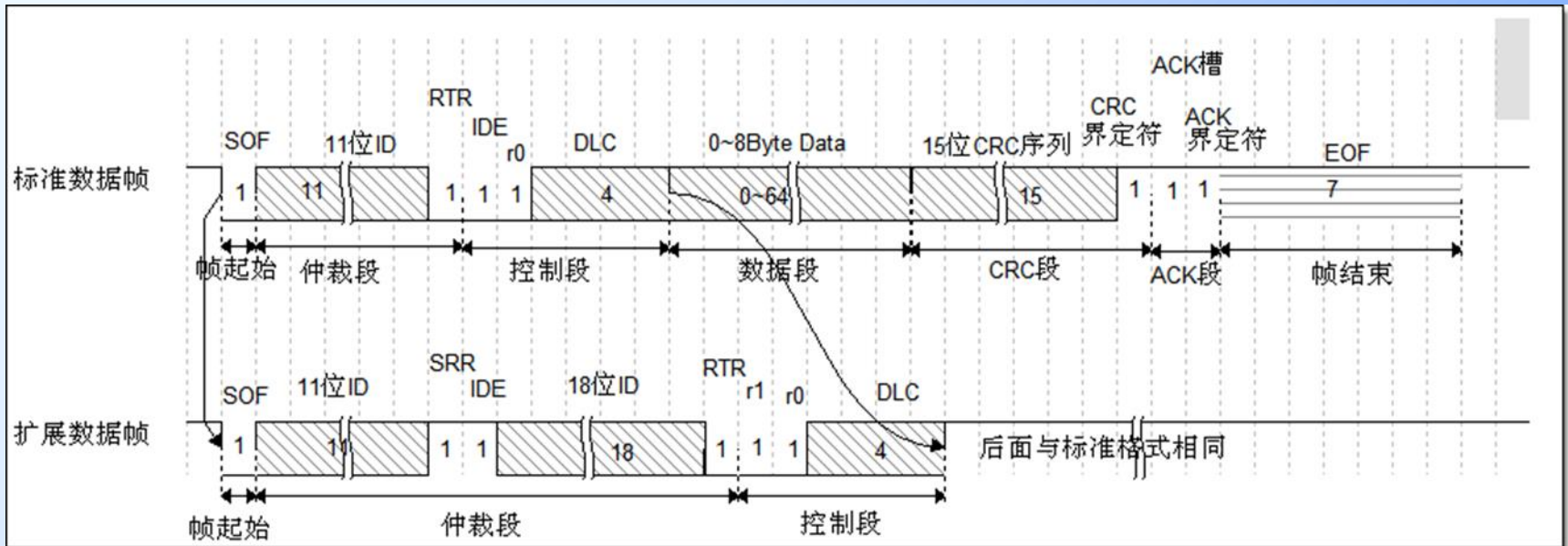
仲裁段

仲裁段ID的优先级也影响着接收设备对报文的反应。因为在CAN总线上数据是以广播的形式发送的，所有连接在CAN总线的节点都会收到所有其它节点发出的有效数据，因而CAN控制器大多具有根据ID过滤报文的功能，它可以控制自己只接收某些ID的报文。

CAN—通讯实验



RTR、IDE、SRR位

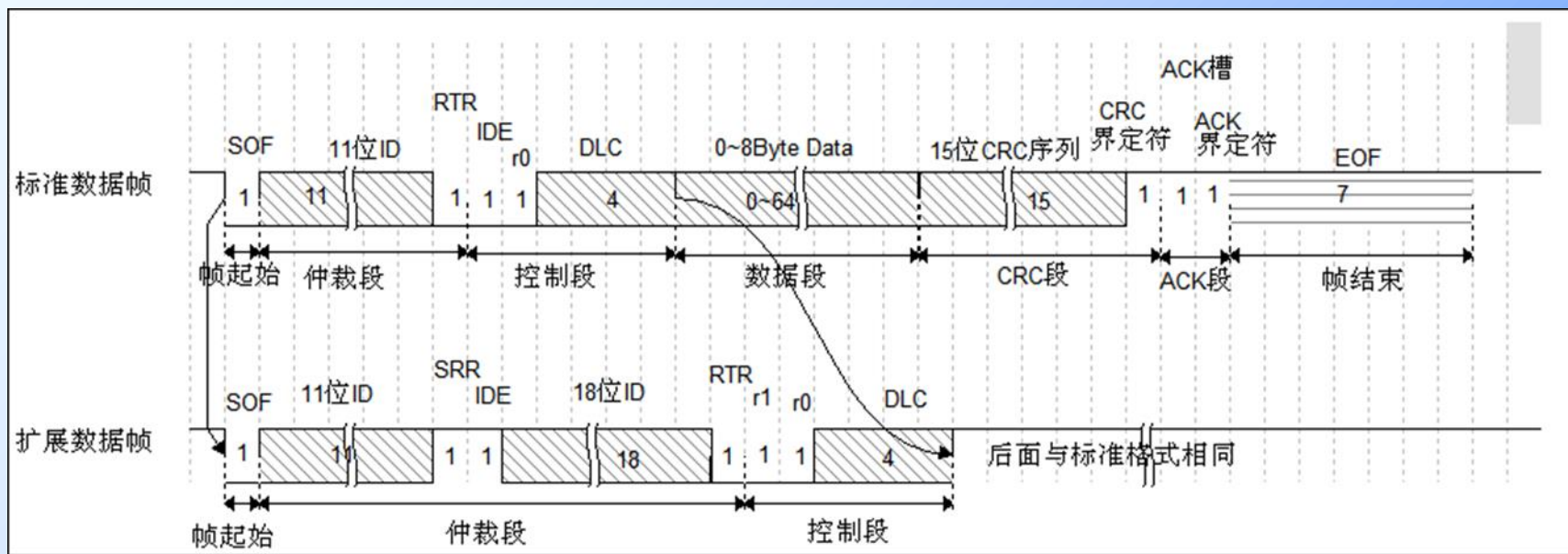


- **RTR位 (Remote Transmission Request Bit)**, 译作远程传输请求位, 它是用于区分数据帧和遥控帧的, 当它为显性电平时表示数据帧, 隐性电平时表示遥控帧。
- **IDE位 (Identifier Extension Bit)**, 译作标识符扩展位, 它是用于区分标准格式与扩展格式, 当它为显性电平时表示标准格式, 隐性电平时表示扩展格式。
- **SRR位 (Substitute Remote Request Bit)**, 只存在于扩展格式, 它用于替代标准格式中的RTR位。由于扩展帧中的SRR位为隐性位, RTR在数据帧为显性位, 所以在两个ID相同的标准格式报文与扩展格式报文中, 标准格式的优先级较高。

CAN—通讯实验



控制段



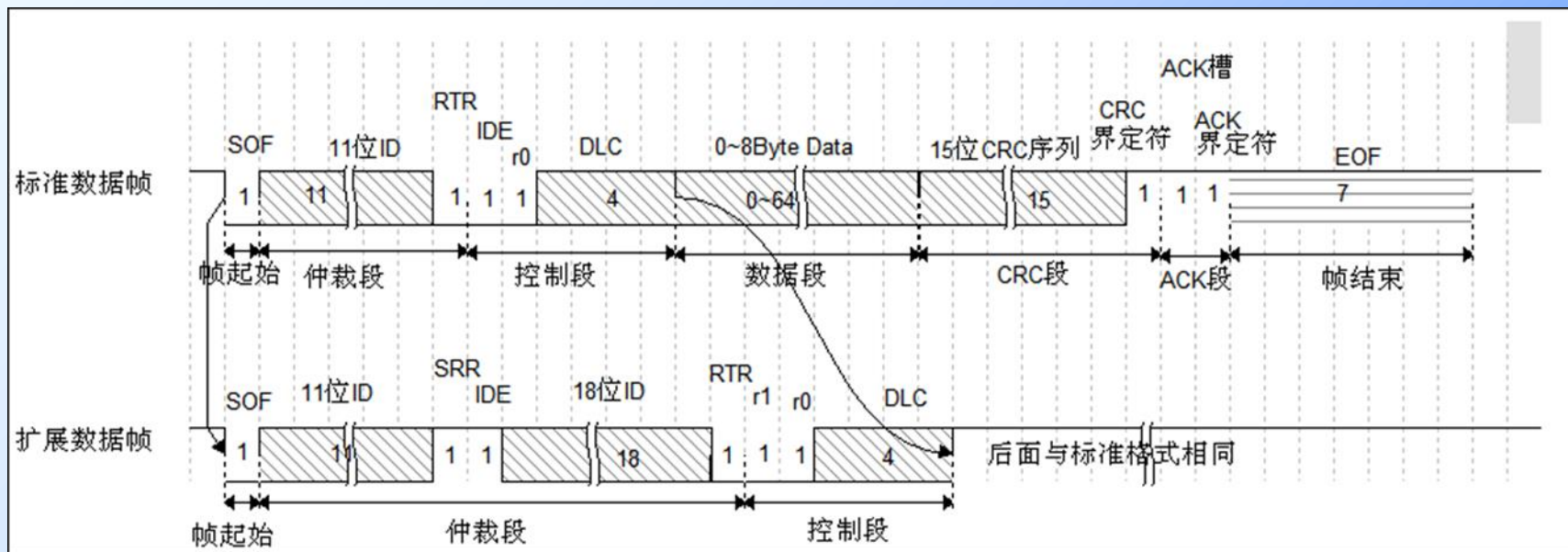
- 控制段

在控制段中的r1和r0为保留位，默认设置为显性位。它最主要的是DLC段(Data Length Code)，译为数据长度码，它由4个数据位组成，用于表示本报文中的数据段含有多少个字节，DLC段表示的数字为0~8。

CAN—通讯实验



数据段



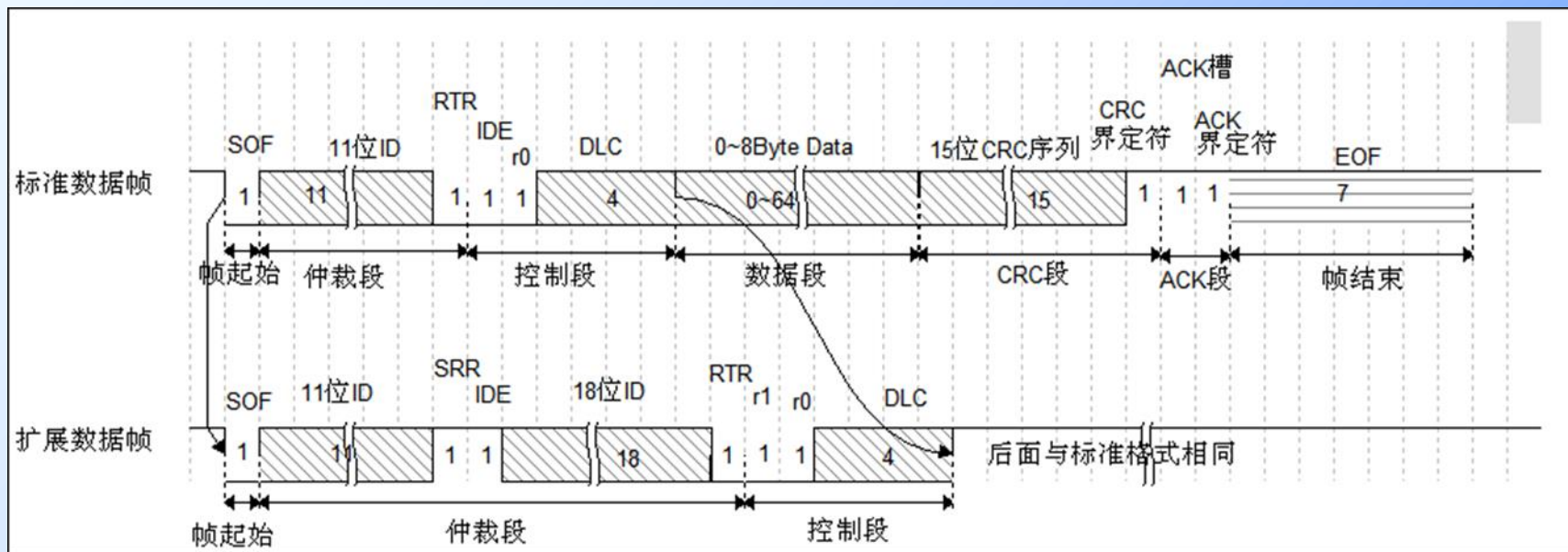
- 数据段

数据段为数据帧的核心内容，它是节点要发送的原始信息，由0~8个字节组成，MSB先行。

CAN—通讯实验



CRC段



- CRC段

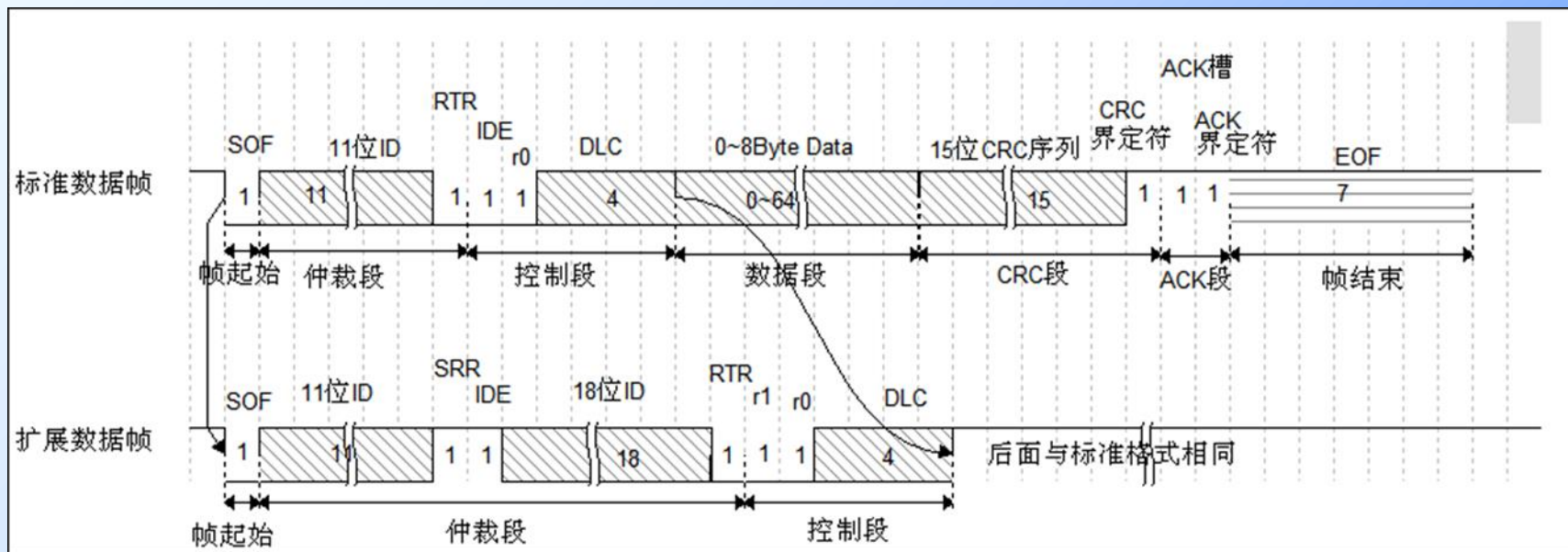
为了保证报文的正确传输，CAN的报文包含了一段15位的CRC校验码，一旦接收节点算出的CRC码跟接收到的CRC码不同，则它会向发送节点反馈出错信息，利用错误帧请求它重新发送。CRC部分的计算一般由CAN控制器硬件完成，出错时的处理则由软件控制最大重发数。

在CRC校验码之后，有一个CRC界定符，它为隐性位，主要作用是把CRC校验码与后面的ACK段间隔起来。

CAN—通讯实验



ACK段



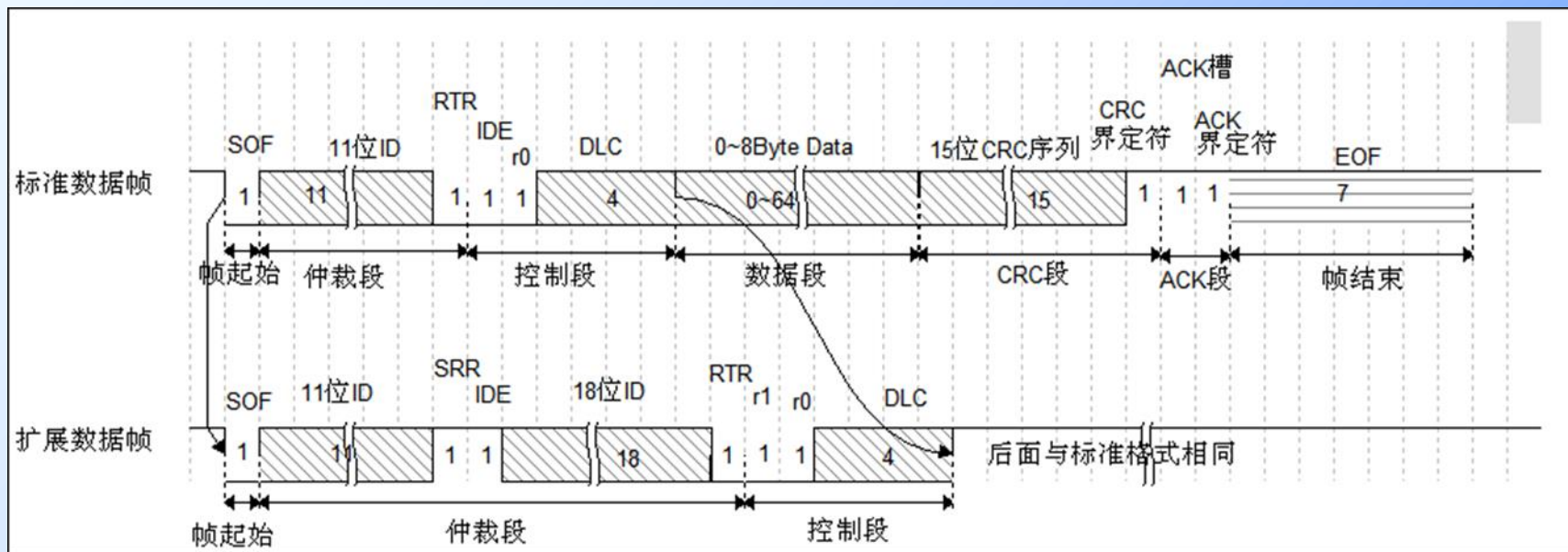
- ACK段

ACK段包括一个ACK槽位，和ACK界定符位。类似I2C总线，在ACK槽位中，发送节点发送的是隐性位，而接收节点则在这一位中发送显性位以示应答。在ACK槽和帧结束之间由ACK界定符间隔开。

CAN—通讯实验



帧结束



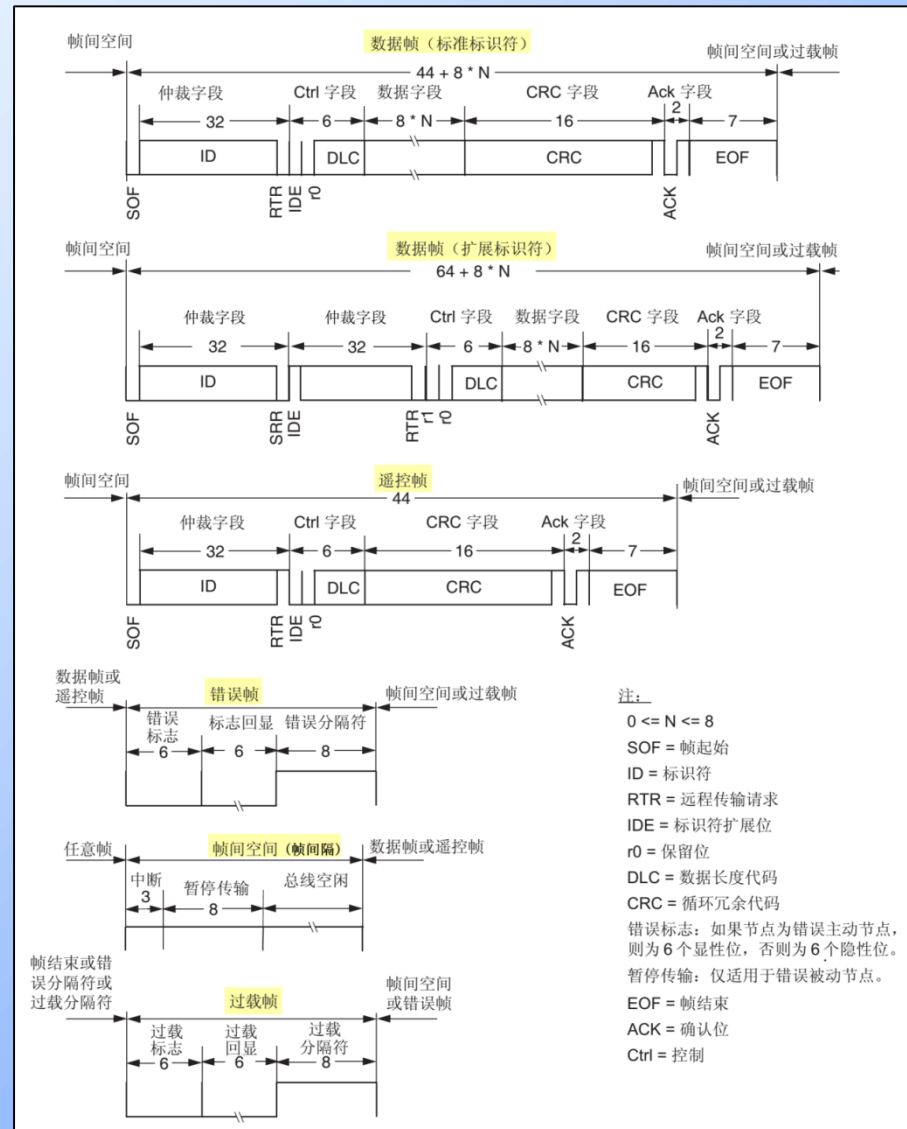
- 帧结束

EOF段(End Of Frame), 译为帧结束, 帧结束段由发送节点发送的7个隐性位表示结束。

CAN—通讯实验



其它报文



零死角玩转STM32



THANKS

论坛：www.firebbs.cn

淘宝：firestm32.taobao.com



扫描进入淘宝店铺