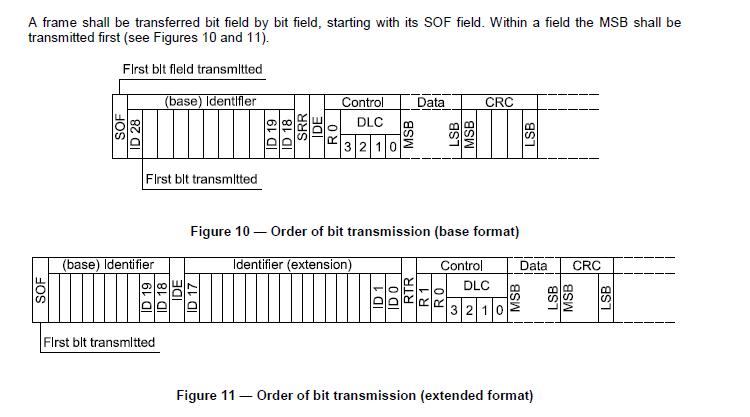
IOS11898-1 中的帧格式说明：

# 一：两种帧ID格式



1: 11位的CAN ID。

帧由  **ID28 - ID1**（11）  +  **SRR** + **IDE** + **Control filed**（只有 **1 位 R0**）+ **Data filed**  + **CRC field**构成。

2:29 位 ID：

**（ID28 - ID18）** +（**IDE**） +（**ID17 - ID0**） + **RTR**+**Control** + **Data** + **CRC**    。

# 二：解释仲裁场

**1：在解释仲裁场之前，先解释一下两个概念，线与和回读,**

线与：

由于can 总线的特殊结构，两根线，如果两个线的电压差超过限定，代表总线数据为**1（又叫隐形位**），如果电**压差低于门限**，就**为0（为显性位**），由于**只要**有个节点**把电压差拉低**，那么**所有的电压差都会降低**，**那么bus上的数据就都为0（所以叫显性位**），那么这么来看，所**有节点都发1**，**总线才为 1**，如果**有一个节点发 0**，**那么就为 0。**所以如果一个节点不想干扰其他节点，**就会往总线发 1.**

回读：

每个can 节点向总线发送数据的时候都会回读，就是自己想总线发送一位数据，也同时读取总线上的数据。如果**发现**与**自己发送的数据不一样**，那么就会停止发送。比如有其他节点发0，而自己发的是1，那么退出发送过程。

:2：**解释完了线与和回读，可以解释一下仲裁场（包含ID 和 RTR）：**

**ID：**这里的**CAN ID** 其实就是**用来仲裁**，判别**优先级高低**，如果**有多个can节点**向总线发送了0，**通知所有节点**，**有节点**要**开始发送数据流**，这个时候，所有节点开始硬件准备（停止发送数据），那么**这些准备发送数据的节点**开始**PK**（**发送自己的ID**），**并且回读数据**，**由于线与**的存在，如果发现**回读出现错误**（**其实就是其他节点发送了0**，**自己发1**），停止发送。当所有的ID都发送完毕后，最后留下来**的肯定是ID最小**的（0 位最多，而且靠前），**该节点获得权限**，发送数据。

**RTR**：用来**区分远程帧**和**数据帧**

**IDE：**Identifier Extension Bit 区分扩展ID和标准ID：

**SRR：**这一位用来在标准帧和扩展帧同时存在的时候，并且扩展帧的前 11 位ID 和标准帧的一样时让标准帧获得仲裁，因为标准帧该位为0，而扩展帧的此位为1（表示后面还有扩展ID），所以标准帧获得控制权。而在标准帧中SRR后面还有一位IDE（为0），也就是说标准帧的前11位和扩展帧前11位一样，而标准帧的SRR和扩展帧的IDE对应。

**ACK：**

包含2位，前一位应答位在发送报文时为隐性，如果总线上有其他节点将报文接收（CRC码正确），那么接收节点会发送显性位0将总线电平改写。后一位是应答场分隔符，隐性.

**EOF：七个隐形位，标志数据发送完毕**。

**远程帧：**

主要是用来向某个固定的节点主动寻求数据，发送该帧，接收方会根据自己的属性，去选择是否发送数据给对应的节点。