

Autosar CAN开发04（从实际应用认识CAN报文）

 **嵌软小白贝** 
汽车制造业 autosar软件工程师

已关注

104 人赞同了该文章 >

建议同时阅读本专栏的：

[CAN开发03（从实际应用认识CAN总线的物理层）](#)

[CAN开发04（从实际应用认识CAN报文）](#)

[CAN开发05（从实际应用认识CAN波特率）](#)

前言

想起当时还没接触CAN的时候，听到别人提到CAN报文、CAN帧这些东西，总是在想，报文？帧？这什么东西。

然后百度：



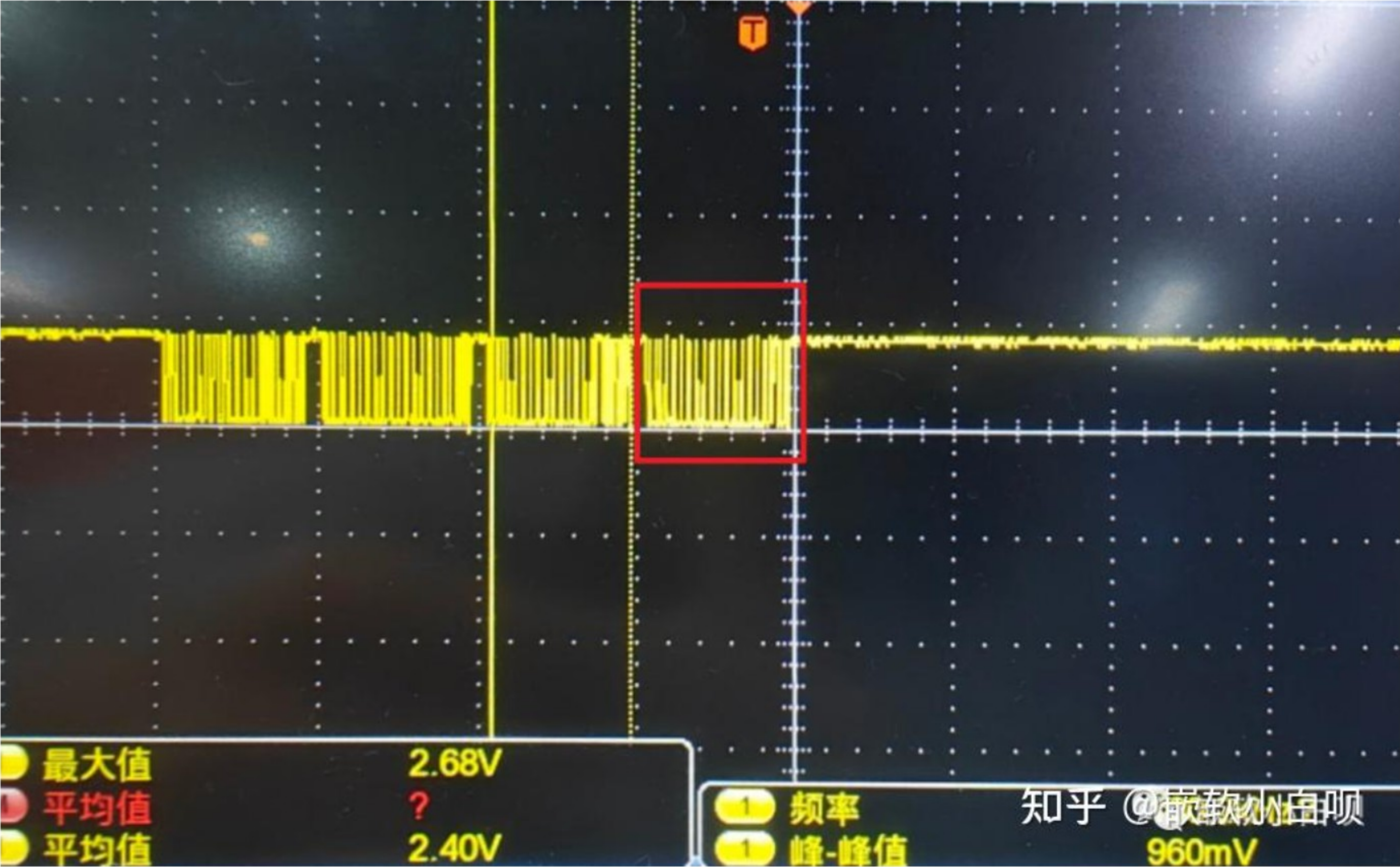
大哥啊，我只是想知道什么是报文，感觉应该不难吧？？？为什么要讲得这么专业，生怕我入行抢饭碗是吧，还是，我太菜了，根本看不懂。。。

...

什么是CAN报文（以标准CAN格式报文为例）

现在，按照我的理解，所谓的一帧CAN报文，通俗来讲就是：**CAN节点发送到CAN总线的一个数据包。**

如下图所示，图中有四帧CAN报文。（用示波器探头测量CANL或CANH）：



对应CAN上位机如下图所示：

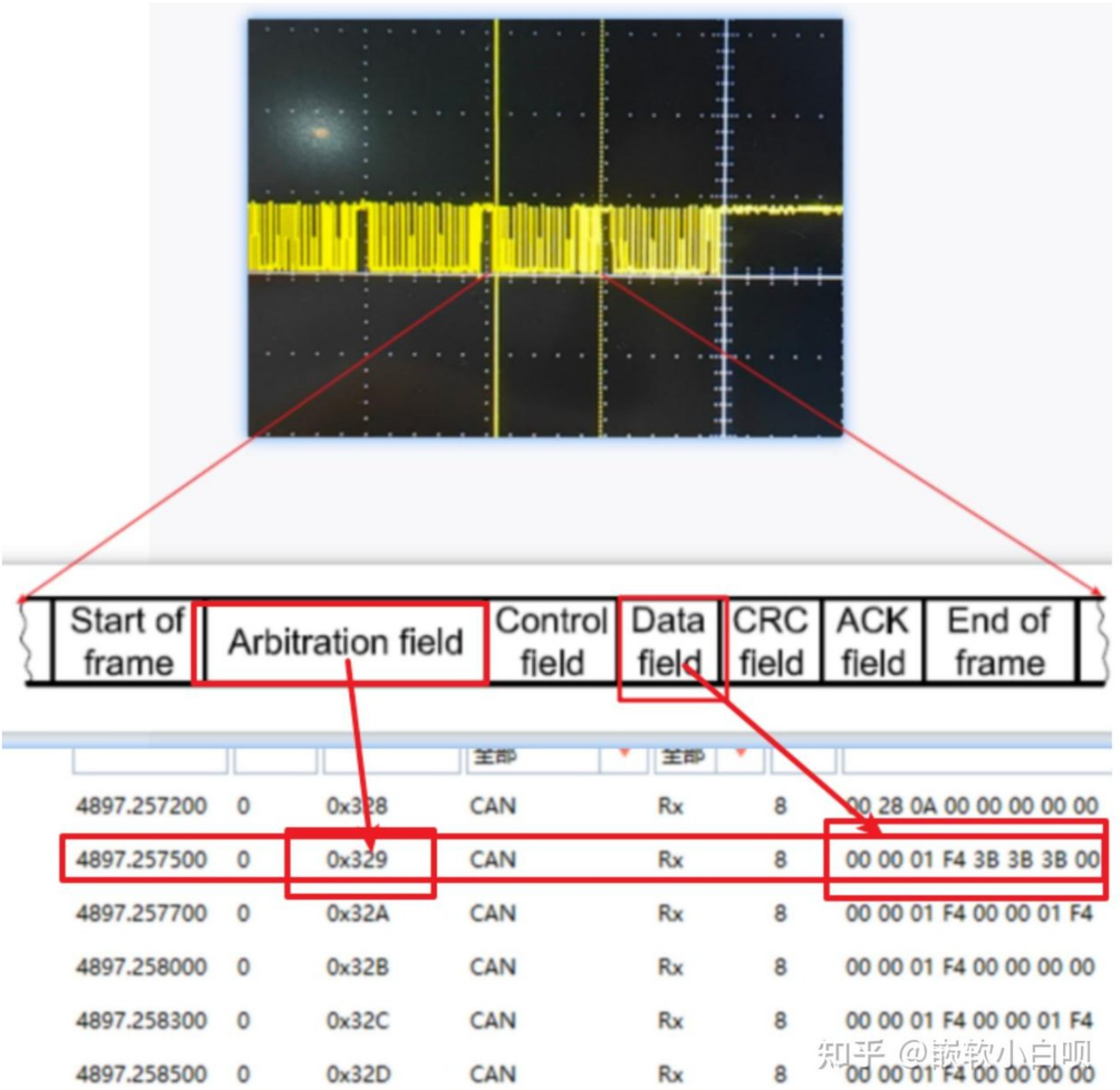


214.073000	1	0x1F8	CAN	Rx	8	00 00 00 00 00 00 0E DF
214.073000	0	0x1F8	CAN	Tx	8	00 00 00 00 00 00 0E DF
214.074000	1	0x1F8	CAN	Rx	8	00 15 70 C4 00 00 00 00

这就是大家所说的一帧CAN报文。

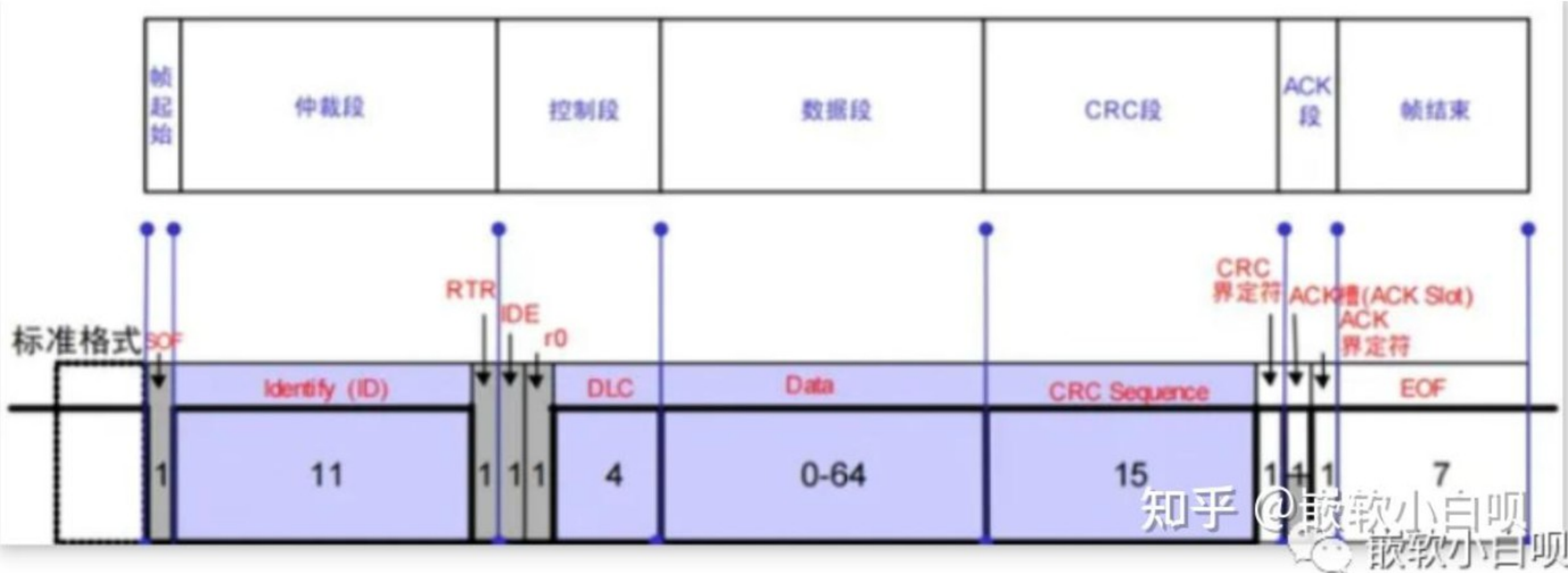
那么，一帧CAN报文的物理结构是怎么样的呢？

如下图所示：



把这3张图一结合，是不是马上对CAN报文有了一个更深刻的认识？

接下来，我们再把CAN报文的结构更细化一些：



(图片来源于网络)

可见，一帧CAN报文的组成还是挺多东西的，但是，对我们平时使用来说，我们只看到两个段：

1、**仲裁段**，也就是这一帧CAN报文的报文ID，

时间戳	标志	ID	名称	方向	长度	数据
4897.257200	0	0x328	CAN	Rx	8	00 28 0A 00 00 00 00 00
4897.257500	0	0x329	CAN	Rx	8	00 00 01 F4 3B 3B 3B 00
4897.257700	0	0x32A	CAN	Rx	8	00 00 01 F4 00 00 01 F4
4897.258000	0	0x32B	CAN	Rx	8	00 00 01 F4 00 00 00 00
4897.258300	0	0x32C	CAN	Rx	8	00 00 01 F4 00 00 01 F4

仲裁段，即报文ID

2、**数据段**，也就是这一帧CAN报文具体传输的数据内容。

时间戳	标志	ID	名称	方向	长度	数据
4897.257200	0	0x328	CAN	Rx	8	00 28 0A 00 00 00 00 00
4897.257500	0	0x329	CAN	Rx	8	00 00 01 F4 3B 3B 3B 00
4897.257700	0	0x32A	CAN	Rx	8	00 00 01 F4 00 00 01 F4
4897.258000	0	0x32B	CAN	Rx	8	00 00 01 F4 00 00 00 00
4897.258300	0	0x32C	CAN	Rx	8	00 00 01 F4 00 00 01 F4
4897.258500	0	0x32D	CAN	Rx	8	00 00 01 F4 00 00 00 00

数据段，即报文数据

其它部分：帧起始、控制段、CRC段、ACK段、帧结束，这些是一帧报文固定有的东西，CAN盒在发送或接收报文时会按照CAN标准协议解析这些地方，对我们现在只是简单认识CAN报文和使用CAN报文，暂时不需要去深究它们。

实际上，我们经常接触的除了上面的标准CAN格式报文，还有另外3种常接触的CAN报文，如下图所示：

	仲裁域长度 (CANID)	数据域长度 (Data)
标准CAN格式	11Bit	0-8Byte
标准CANFD格式	11Bit	0-64Byte
扩展CAN格式	29Bit	0-8Byte
扩展CANFD格式	29Bit	0-64Byte

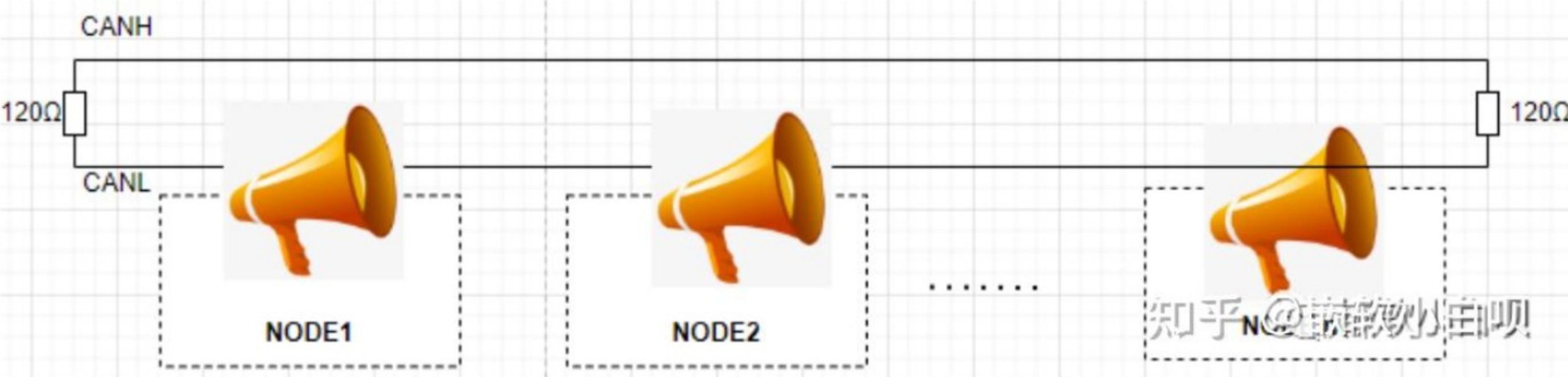
表中列出来的是平时应用过程中，对不同CAN格式报文我们需要知道的关键差异，对入门来说，只要知道这些就够了。

知道了什么是CAN报文，接下来，就要去看看CAN报文具体是怎么发到CAN总线了。

CAN报文的在CAN总线上的传输

我们要知道，**CAN数据传输方式是广播式的**。也就是一对多，一个CAN节点发的CAN报文，其它CAN节点都能够接收到。

就像很多人在同一个房间一样，每个人什么时候都能说话，而且房间是正常大小，只要一个人正常说话，其他人都能听到。

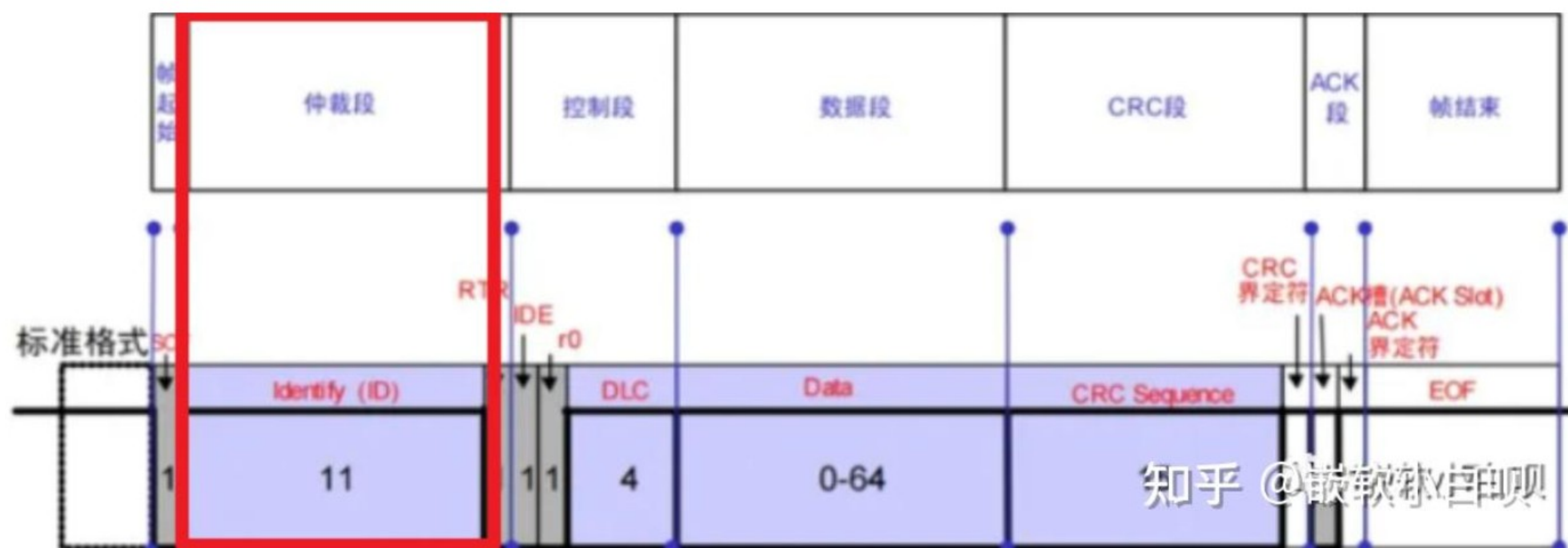


由于CAN报文在总线上是这样的传输方式。

因此，这里马上就能想到一个问题：如果两个节点同时往外发报文怎么办？

CAN总线的仲裁机制

所以，CAN总线如果出现多个CAN节点同时发送报文时，会有仲裁机制。



仲裁发生的位置就在CAN报文的仲裁段（报文ID）上。

关于仲裁，我们这里有个结论：报文ID越小，优先级越高。

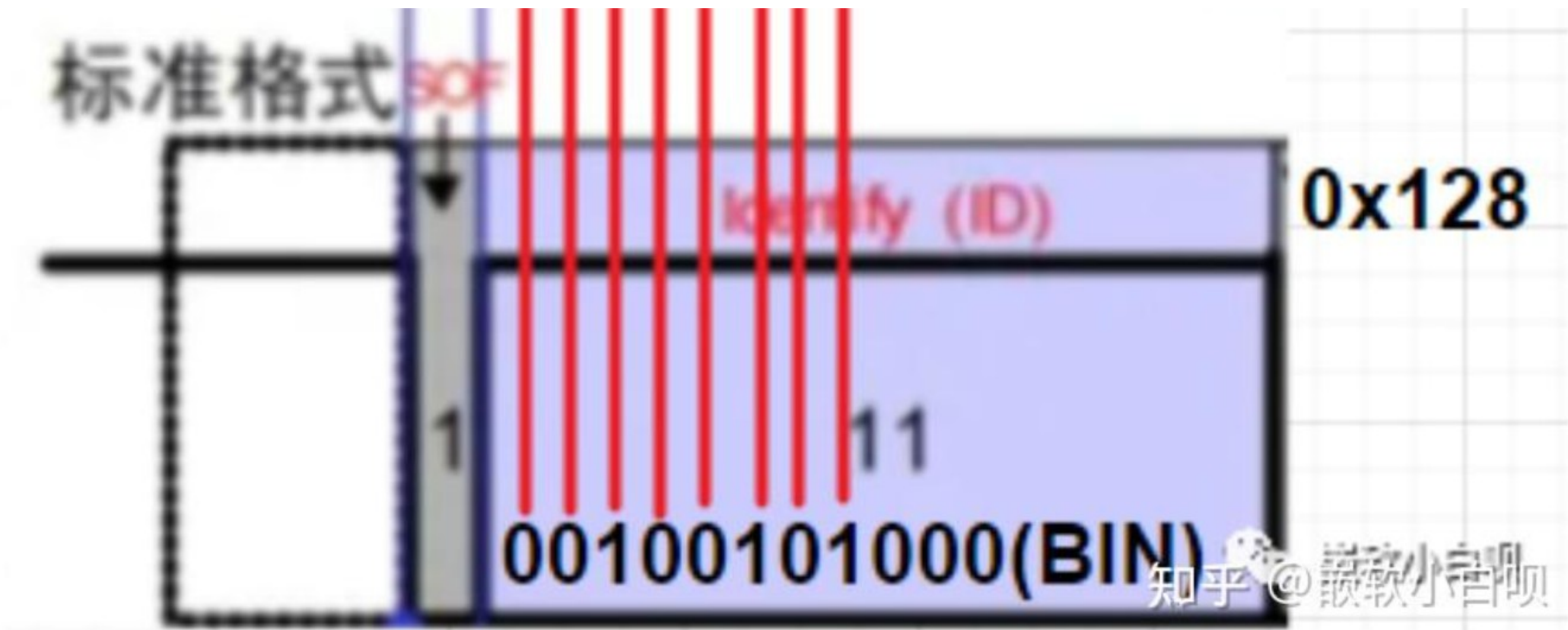
为什么呢？

我们先认识一下“线与”的概念：显性覆盖隐性⁺，当电路通路时，有一个端点接地（GND），那么整个电路电压就为零了。

因此：（逻辑电平0） & （逻辑电平1） = （逻辑电平0）

所以，我们举个栗子就能理解为什么仲裁机制中，CANID越小，优先级越高了，如下图所示：两个CAN控制器同时发送CAN报文，当CANID发到第8位时，由于0x123的第8位是0，0x128的第8位是1，因此，CAN总线表现为逻辑0，0x123报文仲裁胜利。



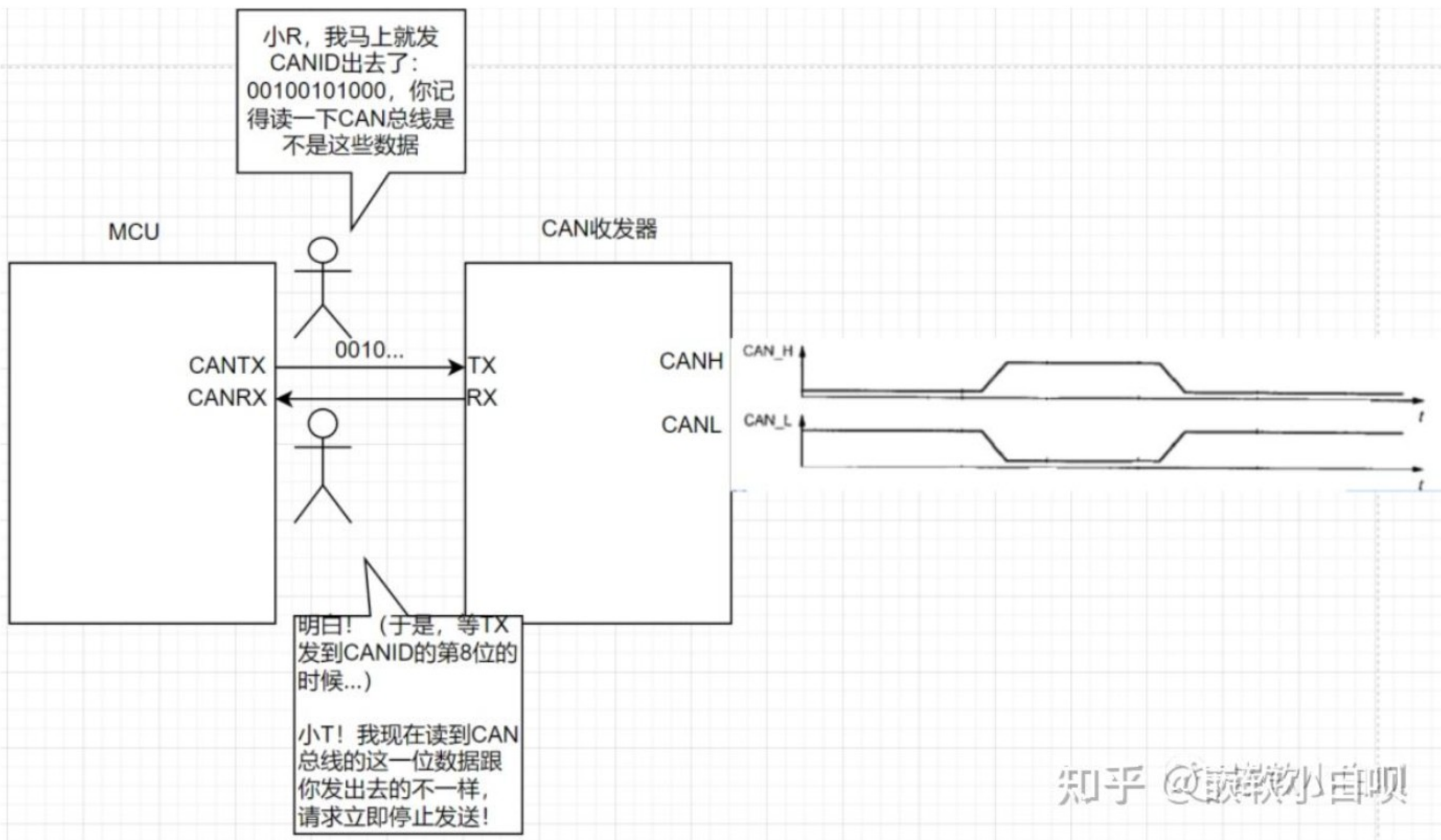


CAN控制器的回读机制+

那么，CAN控制器怎么就知道自己仲裁成功还是仲裁失败了？

所以，CAN控制器还有一个机制：发出数据时，会进行回读。

简单理解，就是CAN控制器通过TX脚发出一帧CAN报文的时候，会通过CAN控制器的RX脚进行回读，如果读到自己发出去的逻辑电平跟自己会读到的逻辑电平不一致，在仲裁段时就会停止发报文，在其它段则产生错误帧。



当发出0x128报文的节点发到CANID的第8位时，它发现，明明自己发出去的是逻辑电平1，但读回来就变成了逻辑电平0，因为总线电平被0x123的第8位拉成0。

于是，它就知道自己跟人家竞争失败了。

仲裁失败的一方，则退出CAN报文的发送，等待仲裁胜利的报文发送完成后再次发送，若继续出现与其它节点同时发送报文的情况，则继续仲裁。

所以，依靠着仲裁机制，CAN总线上即便很多CAN节点接上去，然后任意发CAN报文都不会有啥问题了。

结束

好了，当认识了什么是CAN报文及CAN报文是怎么样在总线上传输的之后，我们下一篇看一下CAN波特率是啥玩意。

建议同时阅读本专栏的：

[CAN开发03（从实际应用认识CAN总线的物理层）](#)

[CAN开发04（从实际应用认识CAN报文）](#)

[CAN开发05（从实际应用认识CAN波特率）](#)

发表日期：2023.12.25

朋友们，关注下我呀，我以我过来人，再用小白的角度认真写的知识总结一定让你的脑子饿肚子进来，扶墙出去...

返回目录

[Autosar BSW 开发笔记（目录）](#)

编辑于 2023-12-25 20:29 · IP 属地广东