

## CAN 学习手记

题记：

在这信息爆炸的年代，学习一样新的知识，尤其是 IT 技术的学习，网络是必不可少的。一阵狂搜过后，发现基于 CAN 总线的介绍确实不少，但是大多好像都雷同了，也许是不不断转贴的结果吧。而且这些知识点也都太零散了，让人看了，尤其是初学者有点找不着北的感觉。所以，本着谦虚好学脚踏实地的菜鸟精神，决定把网上搜的，书上看的，再添油加醋的做一番比较系统的整理。算是为学习 CAN 总线理清思路，快速上手打基础吧。

### 一、什么是 CAN 总线？

CAN，全称为“Controller Area Network”，即控制器局域网，是国际上应用最广泛的现场总线之一。最初，CAN 被设计作为汽车环境中的微控制器通讯，在车载各电子控制装置 ECU 之间交换信息，形成汽车电子控制网络。比如：发动机管理系统、变速箱控制器、仪表装备、电子主干系统中，均嵌入 CAN 控制装置。

一个由 CAN 总线构成的单一网络中，理论上可以挂接无数个节点。实际应用中，节点数目受网络硬件的电气特性所限制。例如，当使用 Philips P82C250 作为 CAN 收发器时，同一网络中允许挂接 110 个节点。CAN 可提供高达 1Mbit/s 的数据传输速率，这使实时控制变得非常容易。另外，硬件的错误检定特性也增强了 CAN 的抗电磁干扰能力。

### 二、CAN 总线是如何发展的？

CAN 最初出现在 80 年代末的汽车工业中，由德国 Bosch 公司最先提出。当时，由于消费者对于汽车功能的要求越来越多，而这些功能的实现大多是基于电子操作的，这就使得电子装置之间的通讯越来越复杂，同时意味着需要更多的连接信号线。提出 CAN 总线的最初动机就是为了解决现代汽车中庞大的电子控制装置之间的通讯，减少不断增加的信号线。于是，他们设计了一个单一的网络总线，所有的外围器件可以被挂接在该总线上。1993 年，CAN 已成为国际标准 ISO11898(高速应用)和 ISO11519（低速应用）。

CAN 是一种多主方式的串行通讯总线，基本设计规范要求有高的位速率，高抗电磁干扰性，而且能够检测出产生的任何错误。当信号传输距离达到 10Km 时，CAN 仍可提供高达 50Kbit/s 的数据传输速率。

由于 CAN 总线具有很高的实时性能，因此，CAN 已经在汽车工业、航空工业、工业控制、安全防护等领域中得到了广泛应用。

**三、CAN 总线是如何工作的？**

CAN 通讯协议主要描述设备之间的信息传递方式。CAN 层的定义与开放系统互连模型（OSI）一致。每一层与另一设备上相同的那一层通讯。实际的通讯发生在每一设备上相邻的两层，而设备只通过模型物理层的物理介质互连。CAN 的规范定义了模型的最下面两层：数据链路层和物理层。下表中展示了 OSI 开放式互连模型的各层。应用层协议可以由 CAN 用户定义成适合特别工业领域的任何方案。已在工业控制和制造业领域得到广泛应用的标准是 DeviceNet，这是为 PLC 和智能传感器设计的。在汽车工业，许多制造商都应用他们自己的标准。

7	应用层	最高层。用户、软件、网络终端等之间用来进行信息交换。 如：DeviceNet
6	表示层	将两个应用不同数据格式的系统信息转化为能共同理解的格式
5	会话层	依靠低层的通信功能来进行数据的有效传递。
4	传输层	两通讯节点之间数据传输控制。操作如：数据重发，数据错误修复
3	网络层	规定了网络连接的建立、维持和拆除的协议。如：路由和寻址
2	数据链路层	规定了在介质上传输的数据位的排列和组织。如：数据校验和帧结构
1	物理层	规定通讯介质的物理特性。如：电气特性和信号交换的解释

表 1 OSI 开放系统互连模型

CAN 能够使用多种物理介质，例如双绞线、光纤等。最常用的就是双绞线。信号使用差分电压传送，两条信号线被称为“CAN\_H”和“CAN\_L”，静态时均是 2.5V 左右，此时状态表示为逻辑“1”，也可以叫做“隐性”。用 CAN\_H 比 CAN\_L 高表示逻辑“0”，称为“显形”，此时，通常电压值为：CAN\_H = 3.5V 和 CAN\_L = 1.5V。

#### 四、CAN 有哪些特性？

- CAN 具有十分优越的特点，使人们乐于选择。这些特性包括：
- 1、低成本；
  - 2、极高的总线利用率；
  - 3、很远的数据传输距离(长达 10Km)；
  - 4、高速的数据传输速率（高达 1Mbit/s）；
  - 5、可根据报文的 ID 决定接收或屏蔽该报文；
  - 6、可靠的错误处理和检错机制；
  - 7、发送的信息遭到破坏后，可自动重发；
  - 8、节点在错误严重的情况下具有自动退出总线的功能；
  - 9、报文不包含源地址或目标地址，仅用标志符来指示功能信息、优先级信息。

#### 五、Philips 制造的 CAN 芯片有哪些？

类别	型号	备注
CAN 微控制器	P87C591	替代 P87C592
CAN 独立控制器	SJA1000	替代 82C200
CAN 收发器	PCA82C250	高速 CAN 收发器
	PCA82C251	高速 CAN 收发器
	PCA82C252	容错 CAN 收发器
	TJA1040	高速 CAN 收发器

	TJA1041	高速 CAN 收发器
	TJA1050	高速 CAN 收发器
	TJA1053	容错 CAN 收发器
	TJA1054	容错 CAN 收发器
LIN 收发器	TJA1020	LIN 收发器

表 2 CAN 芯片一览表

## 六、CAN 总线如何进行位仲裁？

CSMA/CD 是“载波侦听多路访问/冲突检测”（Carrier Sense Multiple Access with Collision Detect）的缩写。

利用 CSMA 访问总线，可对总线上信号进行检测，只有当总线处于空闲状态时，才允许发送。利用这种方法，可以允许多个节点挂接到同一网络上。当检测到一个冲突位时，所有节点重新回到‘监听’总线状态，直到该冲突时间过后，才开始发送。在总线超载的情况下，这种技术可能会造成发送信号经过许多延迟。为了避免发送时延，可利用 CSMA/CD 方式访问总线。当总线上有两个节点同时进行发送时，必须通过“无损的逐位仲裁”方法来使有最高优先权的报文优先发送。在 CAN 总线上发送的每一条报文都具有唯一的一个 11 位或 29 位数字的 ID。CAN 总线状态取决于二进制数‘0’而不是‘1’，所以 ID 号越小，则该报文拥有越高的优先权。因此一个为全‘0’标志符的报文具有总线上的最高级优先权。可用另外的方法来解释：在消息冲突的位置，第一个节点发送 0 而另外的节点发送 1，那么发送 0 的节点将取得总线的控制权，并且能够成功的发送出它的信息。

## 七、CAN 的高层协议

CAN 的高层协议（也可理解为应用层协议）是一种在现有的底层协议（物理层和数据链路层）之上实现的协议。高层协议是在 CAN 规范的基础上发展起来的应用层。许多系统（像汽车工业）中，可以特别制定一个合适的应用层，但对于许多的行业来说，这种方法是不经济的。一些组织已经研究并开放了应用层标准，以使系统的综合应用变得十分容易。

一些可使用的 CAN 高层协议有：

- 1、制定组织主要高层协议
- 2、CiA CAL 协议
- 3、CiA CANOpen 协议
- 4、ODVA DeviceNet 协议
- 5、Honeywell SDS 协议
- 6、Kvaser CANKingdom 协议

## 八、什么是标准格式 CAN 和扩展格式 CAN?

标准 CAN 的标志符长度是 11 位，而扩展格式 CAN 的标志符长度可达 29 位。CAN 协议的 2.0A 版本规定 CAN 控制器必须有一个 11 位的标志符。同时，在 2.0B 版本中规定，CAN 控制器的标志符长度可以是 11 位或 29 位。遵循 CAN2.0B 协议的 CAN 控制器可以发送和接收 11 位标识符的标准格式报文或 29 位标识符的扩展格式报文。如果禁止 CAN2.0B，则 CAN 控制器只能发送和接收 11 位标识符的标准格式报文，而忽略扩展格式的报文结构，但不会出现错误。

目前，Philips 公司主要推广的 CAN 独立控制器均支持 CAN2.0B 协议，即支持 29 位标识符的扩展格式报文结构。

## 九、CAN 的报文格式：

（关于这部分内容，对于我下一步要进行编程实验控制 CAN 总线是很重要的一个知识点。很郁闷的是搜了老半天只看见有文字叙述这部分内容的，不仅说得不系统完整，而且远没有贴图说明来的具体直观。不过还好，搜到了一份 E 文版的 CAN 总线资料，再把这个资料详细看了下，再做了整理，还是用图来说明 CAN 传输的报文格式比较容易理解。老外写的东西确实比国内的好，这个不是我吹出来的，有兴趣下了看看就知道了，见附件下载。）

CAN 通信是一种点对多点的传输协议，不是基于地址的传统的点对点传输协议。当一个点传输数据时，总线上的其它点都可以为接受方，它们可以通过 ID 来作出对总线上传送数据的处理（接收或者丢弃）。并且当数据被正确接收到以后，接收方便会作出应答响应。CAN 协议还有一个很实用的功能，就是总线上的



图 2 标准数据帧格式

图 3 扩展数据帧格式

由图 2 和图 3 可以看到数据帧的两种不同格式，两种格式的差别只是它们的 ID 位数不同。

- 1、帧起始。
- 2、仲裁场，仲裁场包括标识符和远程发送请求位（RTR）。

对于 CAN2.0A 标准，标识符的长度为 11 位。RTR 位在数据帧中必须是显性位，而在远程帧必须为隐性位。

对于 CAN2.0，标准格式和扩展格式的仲裁场不同。在标准格式中，仲裁场由 11 位标识符和远程发送请求位组成。在扩展格式中，仲裁场由 29 位标识符和替代远程请求位（SRR）、标志位（IDE）和远程发送请求位组成。

仲裁场的作用之一是说明数据帧或远程帧发送目的地；之二是指出是数据帧还是远程帧。

3、控制场，控制场由 6 个位组成，说明数据帧中有效数据的长度。标准帧的最高位是 IDE 位，扩展帧的最高位是保留位 RB1，它们的次高位都是保留位 RB0。低四位是 DLC (Data Length Code) 位，标识传送的数据字节数 (0-8 字节)。

4、数据场，数据场由数据帧中的发送数据组成。它可以为 0-8 个字节。

5、CRC 场，CRC 场包括 CRC 序列，这部分由 SJA1000 控制芯片完成。

6、应答场，应答场长度为两个位，包括应答间隙和应答界定符。由 SJA1000 控制芯片自动完成。

7、帧结束，每一个数据帧和远程帧均由一标志序列界定，这个标志序列由 7 个“隐性”位组成。这部分由 SJA1000 控制芯片自动完成。

仲裁场、控制场、数据场由软件编程配置 SJA1000 完成；帧起始、CRC 场、应答场、帧结束由 CAN 总线控制芯片 SJA1000 自动完成。

## 十、CAN 的数据错误检测：

不同于其它总线，CAN 协议不能使用应答信息。事实上，它可以将发生的任何错误用信号发出。CAN 协议可使用五种检查错误的方法，其中前三种为基于报文内容检查。

### 1、循环冗余检查 (CRC)

在一帧报文中加入冗余检查位可保证报文正确。接收站通过 CRC 可判断报文是否有错。

### 2、帧检查

这种方法通过位场检查帧的格式和大小来确定报文的正确性，用于检查格式上的错误。

### 3、应答错误

如前所述，被接收到的帧由接收站通过明确的应答来确认。如果发送站未收到应答，那么表明接收站发现帧中有错误，也就是说，ACK 场已损坏或网络中的报文无站接收。CAN 协议也可通过位检查的方法探测错误。

### 4、总线检测

有时，CAN 中的一个节点可监测自己发出的信号。因此，发送报文的站可以观测总线电平并探测发送位和接收位的差异。



## 5、位填充

一帧报文中的每一位都由不归零码表示,可保证位编码的最大效率。然而,如果在一帧报文中太多相同电平的位,就有可能失去同步。为保证同步,同步沿用位填充产生。在五个连续相等位后,发送站自动插入一个与之互补的补码位;接收时,这个填充位被自动丢掉。例如,五个连续的低电平位后,CAN自动插入一个高电平位。CAN通过这种编码规则检查错误,如果在一帧报文中6个相同位,CAN就知道发生了错误。

如果至少有一个站通过以上方法探测到一个或多个错误,它将发送出错标志终止当前的发送。这可以阻止其它站接收错误的报文,并保证网络上报文的一致性。当大量发送数据被终止后,发送站会自动地重新发送数据。作为规则,在探测到错误后23个位周期内重新开始发送。在特殊场合,系统的恢复时间为31个位周期。

但这种方法存在一个问题,即一个发生错误的站将导致所有数据被终止,其中也包括正确的数据。因此,如果不采取自监测措施,总线系统应采用模块化设计。为此,CAN协议提供一种将偶然错误从永久错误和局部站失败中区别出来的办法。这种方法可以通过对出错站统计评估来确定一个站本身的错误并进入一种不会对其它站产生不良影响的运行方法来实现,即站可以通过关闭自己来阻止正常数据因被错误地当成不正确的数据而被终止。

## 6、CAN 可靠性

为防止汽车在使用寿命期内由于数据交换错误而对司机造成危险,汽车的安全系统要求数据传输具有较高的安全性。如果数据传输的可靠性足够高,或者残留下来的数据错误足够低的话,这一目标不难实现。从总线系统数据的角度看,可靠性可以理解为,对传输过程产生的数据错误的识别能力。

残余数据错误的概率可以通过对数据传输可靠性的统计测量获得。它描述了传送数据被破坏和这种破坏不能被探测出来的概率。残余数据错误概率必须非常小,使其在系统整个寿命周期内,按平均统计时几乎检测不到。计算残余错误概率要求能够对数据错误进行分类,并且数据传输路径可由一模型描述。如果要确定CAN的残余错误概率,我们可将残留错误的概率作为具有80~90位的报文传送时位错误概率的函数,并假定这个系统中有5~10个站,并且错误率为1/1000,那么

最大位错误概率为  $10^{-13}$  数量级。例如, CAN 网络的数据传输率最大为 1Mbps, 如果数据传输能力仅使用 50%, 那么对于一个工作寿命 4000 小时、平均报文长度为 80 位的系统, 所传送的数据总量为  $9 \times 10^{10}$ 。在系统运行寿命期内, 不可检测的传输错误的统计平均小于  $10^{-2}$  量级。换句话说, 一个系统按每年 365 天, 每天工作 8 小时, 每秒错误率为  $0.7$  计算, 那么按统计平均, 每 1000 年才会发生一个不可检测的错误。