航芯技术分享 | 一文读懂汽车 CAN 总线技术原理（下）

随着汽车工业的不断发展，汽车电子控制单元逐渐增多，各电控单元之间的信号交换更为复杂。而 CAN 总线可将汽车内部各电控单元之间连接成一个局域网络，实现了信息的共享，大大优化了整车的布线。

接下来，我们将继续为大家分享 CAN 相关技术知识。

CAN 的分层架构

它由三层组成，即应用层、数据链路层和物理层。

- 应用层：该层与操作系统或 CAN 设备的应用程序交互。
- 数据链路层：它在发送、接收和验证数据方面将实际数据连接到协议。
- 物理层：它代表实际的硬件，即 CAN 控制器和收发器。

CAN 物理层特性

CAN 物理层被分为三个部分：在 CAN 控制器芯片中实现的物理编码，指定收发器特性的物理介质附件，物理介质依赖子层，这是特定的应用，没有标准化。

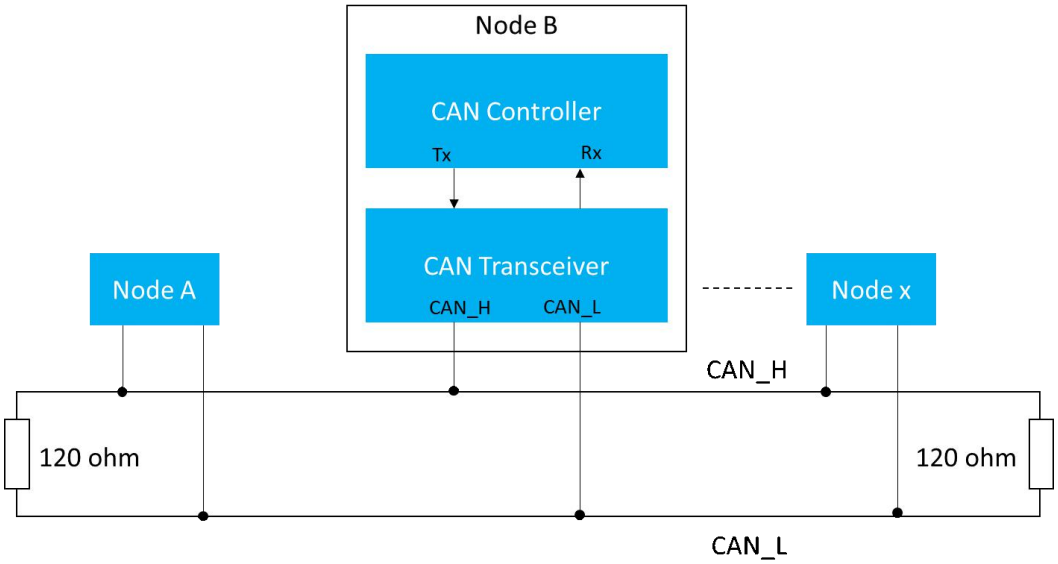

图 1. CAN 总线接线图

物理编码子层

PCS 包括比特编码和解码、比特定时。它为收发器芯片提供连接单元接口，并包含 Tx 和 Rx 引脚，位级错误也通过位填充来处理。

位时序

出于时序目的，CAN 总线上的每个位都划分成至少 4 个时间份额，时间份额逻辑上划分成 4 段：

1. 同步段
2. 传播段
3. 相位缓冲段 1
4. 相位缓冲段 2

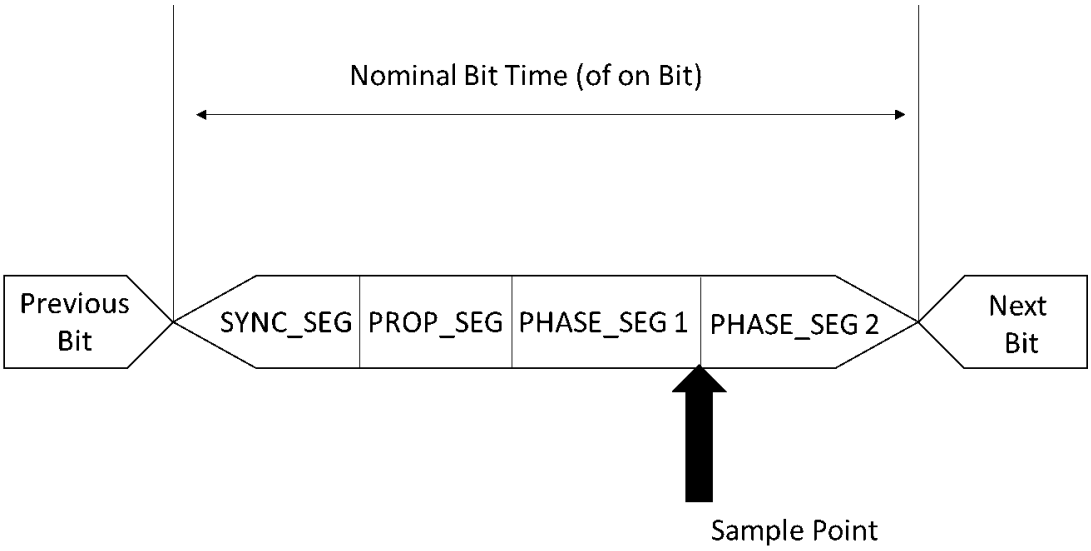

图 2. CAN 位时序

同步段

1 个时间量子长度，用于多个连接在总线上的单元，通过此段实现时序调整，当总线电压电平发生变化（显性到隐性或隐性到显性）时，预计该段会出现位沿。

传播段

用于补偿网络上节点之间的物理延迟，包括发送单元的输出延迟、总线上信号的传播延迟、接收单元的输入延迟。

相位缓冲段

相位缓冲段用于补偿节点间的晶振误差 ,又分为相位缓冲段 1(PS1)和相位缓冲段 2(PS2), 在这个时间段的末端进行总线状态的采样。两个相位缓冲段 PS1 和 PS2 用于补偿总线上的边沿相位误差。

采样点

采样点是位时间内的一个时间点，在该时间点，读取总线电平并进行分析。位时间内的采样点决定 CAN 总线电压是隐性还是显性。以位时间的百分比表示，位置从位时间的起点开始计算，位于阶段 1 和阶段 2 之间。

处理位级错误

位数填充

CAN 协议遵循 NRZ 编码进行传输。逻辑电平在位间隔之间不发生变化。CAN 需要一个逻辑电平的转换来进行再同步。因此，在 5 个相同的连续比特之后，将发送 1 个相反逻辑电平的比特。这就是所谓的东西位，接收器可以识别它。

位错误

一个正在发送比特的节点总是监控总线，如果发射器发送的比特与总线上的比特值不同，则会产生一个错误帧。

物理介质依赖子层

该层在 CAN 收发器芯片中实现，通过 Tx 和 Rx 引脚从 CAN 控制器获得输入，输出驱动 CANH 和 CANL 线。收发器负责不同的比特率，CAN 总线速度指的是 CAN 总线通信速率。最大的 CAN 总线通信速率是 1Mbit/sec。对于特殊的应用，一些 CAN 控制器将处理更高的速度，超过 1Mbit/sec。低速的 CAN 通信速率是 125kbits/sec。

与介质有关的子层

依赖介质的子层是高度特定的应用，不同连接器的引脚分配标准化属于这一层，各种连接器为 DB9、OBD II。

CAN 总线 DB9 引脚布局

CAN 总线通常通过连接器访问。

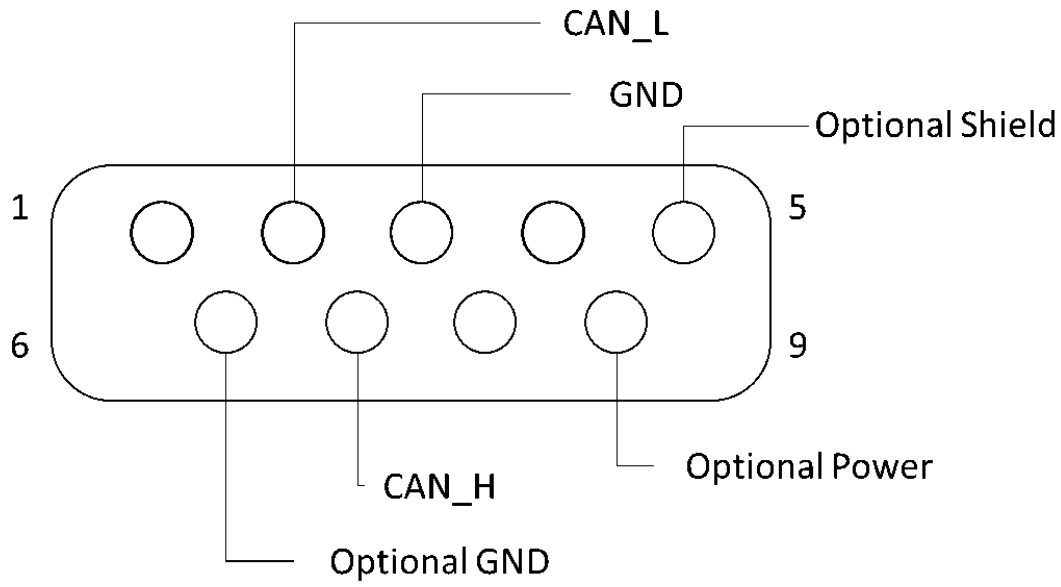

图 3. CAN 总线 DB9 引脚分配

引脚 1：无定义

引脚 2：CAN_L

引脚 3：CAN GND

引脚 4：无定义

引脚 5：CAN_SHLD

引脚 6：GND

引脚 7：CAN_H

引脚 8：无定义

引脚 9：CAN_V+

各种微控制器中的 CAN 总线支持

微控制器应具有 CAN 硬件和软件，提供 CAN 驱动程序以实现通信。Python-CAN 库也可用于为微控制器的硬件组件提供抽象的驱动程序，并用于通过 CAN 网络发送和接收消息。Python CAN 总线也用于测试硬件和 CAN 总线数据记录。

用于 Arduino 的 CAN 总线屏蔽

• CANbus Shield 采用带有 SPI 接口和 CAN 收发器的 CAN 总线控制器，为 Arduino 提供 CAN 总线能力。

• 带有 CAN 总线的 Arduino 有助于从 ECU 获取车速、油耗、温度等信息。

• Arduino CAN 库用于通过 CAN 总线发送和接收 CAN 消息。

树莓派 CAN 总线：

树莓派没有特定的硬件，即 CAN 控制器和 CAN 收发器来支持 CAN 协议。树莓派软件不支持 CAN 总线，树莓派支持通过 SPI 接口进行 CAN 通信。

树莓派通过 SPI 接口连接到板子支持的外部 CAN 控制器，CAN 控制器通过 Rx 和 Tx 线连接到 CAN 收发器。

CAN 控制器示例：SJA100、MCP2515

CAN 收发器示例：TJA1040、MCP2551

ACM32 CAN 总线：

ACM32-F0/F4 芯片内置 1 路~2 路 CAN 控制器，并提供对应的 CAN 总线接口驱动库，搭配外部的 CAN 收发器，保证 CAN 总线数据通讯的安全可靠。

如何读取 CAN 总线数据？

当 CAN 总线与 Microchip CAN 总线分析仪、CAN 总线 Wire Shark 等外部工具连接时，可以通过 CAN USB 适配器访问 CAN 总线数据，该适配器提供与计算机或 PC 的 USB 端口的即时连接。CAN USB 适配器也可以通过以太网、互联网、内联网从任何地方进行控制。CAN 总线 Wireshark 是一种用于 Linux 系统的工具，尤其以以太网网络分析而闻名，它通过使用 SocketCAN 来显示 CAN 消息，SocketCAN 是一组驱动程序和网络堆栈，因此被称为 Linux CAN 总线。CAN to USB 帮助外部工具从 CAN 网络获取消息，然后用于监控和调试接收或传输信息的工具。

但是这些消息是原始格式的。因此，从这些数据记录器收集的数据使用 CAN 总线解码器转换为按比例缩放的工程值。从数据记录器收集的数据也可以存储在 SD 卡中，这有助于控制车辆设置以提高效率。收集的 CAN 总线数据可用于车队管理、研发、诊断等。

用万用表测试 CAN 总线

测试是必要的，以检查任何发生的 CAN 总线故障，如布线、ECU、CAN 网络中的任何一个组件的电压供应故障。CAN 总线的故障排除，如在 CAN 总线线路的物理端添加 120 欧姆的终端电阻，可以诊断出问题。通过用万用表测试，确保终端电阻是 120 欧姆，而且电阻是合适的，没有断裂，还可以通过将万用表切换到交流电压来测试传输的数据。

如何判断汽车是否有 CAN 总线？

配备 CAN 总线的车辆包含 CAN 总线 LED 和 CAN-BUS HID 套件。CAN 总线 LED 与汽车高级系统通信，当此 LED 关闭时，车辆会发出警告。CAN BUS HID KIT 充当 DC 到 AC 转换器，并有助于在最初使用高压电流打开灯。一旦灯启动，它需要较低的电压电流。但是当 HID 使用低功率时，CAN 总线系统会假定灯已关闭并发出警告。为了避免这种情况，使用了 HID 转换套件，它与 CAN 总线系统通信以告知有一个工作灯泡。这些警告告诉我们汽车配备了 CAN 总线。

CAN 总线黑客攻击

CAN 总线黑客攻击是对消费者的威胁。CAN 总线车辆采用了许多无线技术，例如蓝牙，用于接听电话或播放音乐。当车载系统接入车内的 CAN 总线并具备 Wi-Fi 连接能力时，黑客很容易获得 CAN 总线接入并能够控制汽车。Wi-Fi 热点在汽车中很流行，这使得知道汽车 IP 地址的人可以跟踪汽车。这导致汽车制造商对 CAN 总线网络上的传输数据进行保护。

了解更多航芯产品&方案：www.aisinochip.com