CAN 串行协议验证

# CAN原理

嵌入式的工程师一般都知道CAN总线广泛应用到汽车中，其实船舰电子设备通信也广泛使用CAN，随着国家对海防的越来越重视，对CAN的需求也会越来越大。CAN（Controller Area Network）即控制器局域网，是一种能够实现分布式实时控制的串行通信网络。CAN传输速度最高到1Mbps，通信距离最远到10km。

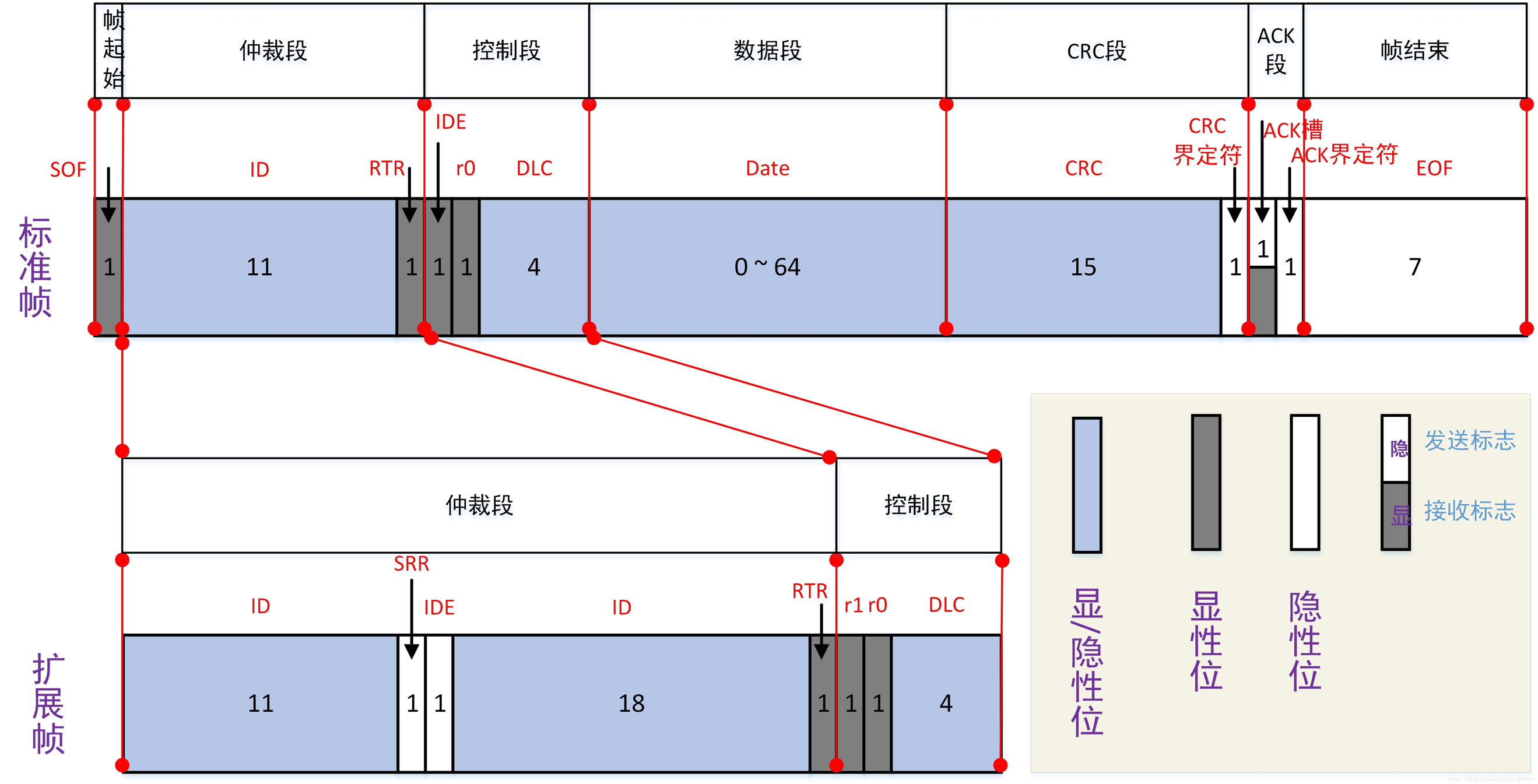
CAN总线传输的是CAN帧：

从帧起始到CRC界定符的位填充规则（只是针对数据帧和远程帧，错误帧和过载帧格式固定）：CAN总线有5个连续相同位后，就插入一个相反位，产生跳变沿，用于同步，消除累计误差。

1. 数据帧

根据仲裁段长度不同分为标准帧（2.0A）和扩展帧（2.0B）。

TR ,/IDE 
111 
CRCEQ 
15 
111 
SRR 
IDE 
64 
111 
RTR rl ro 



* **标准帧：**

1. 帧起始位：0（显电平开始）
2. 仲裁段：

* ID : 11 位，最小优先级越高
* RTR: 1位，显性电平0

1. 控制段

* IDE: 1位，显性电平 0
* r0: 1位，0
* DLC: 4位，数据段的长度码

1. 数据段

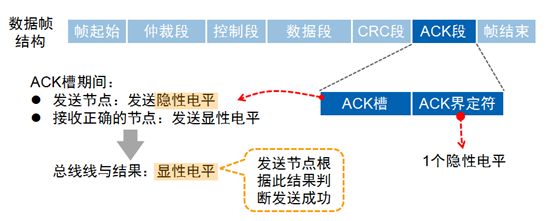
* Data: 0-64bit, byte0~byte7，传输时MSB先传输

1. CRC段

* CRC 校验值：15位，
* CRC界定符：1位

 浈 
帧 起 始 ] 仰 裁 履 0 却 瑷 《 
襖 2 隊 以 
多 项 式 刈 
， ， ， 4 x10 · X 《 · 邓 噲 X 《 · 
數 」 0 《 0 《 
CRC 僨 [ 01 
ACK* 
占 lb•t ， 
0 性 电 平 填 允 

1. ACK段



1. 帧结束：1111111

* **扩展帧：**

1. 起始：0
2. 仲裁段：

* ID: 11位＋SRR（0）+IDE（1）+18位
* RTR: 0

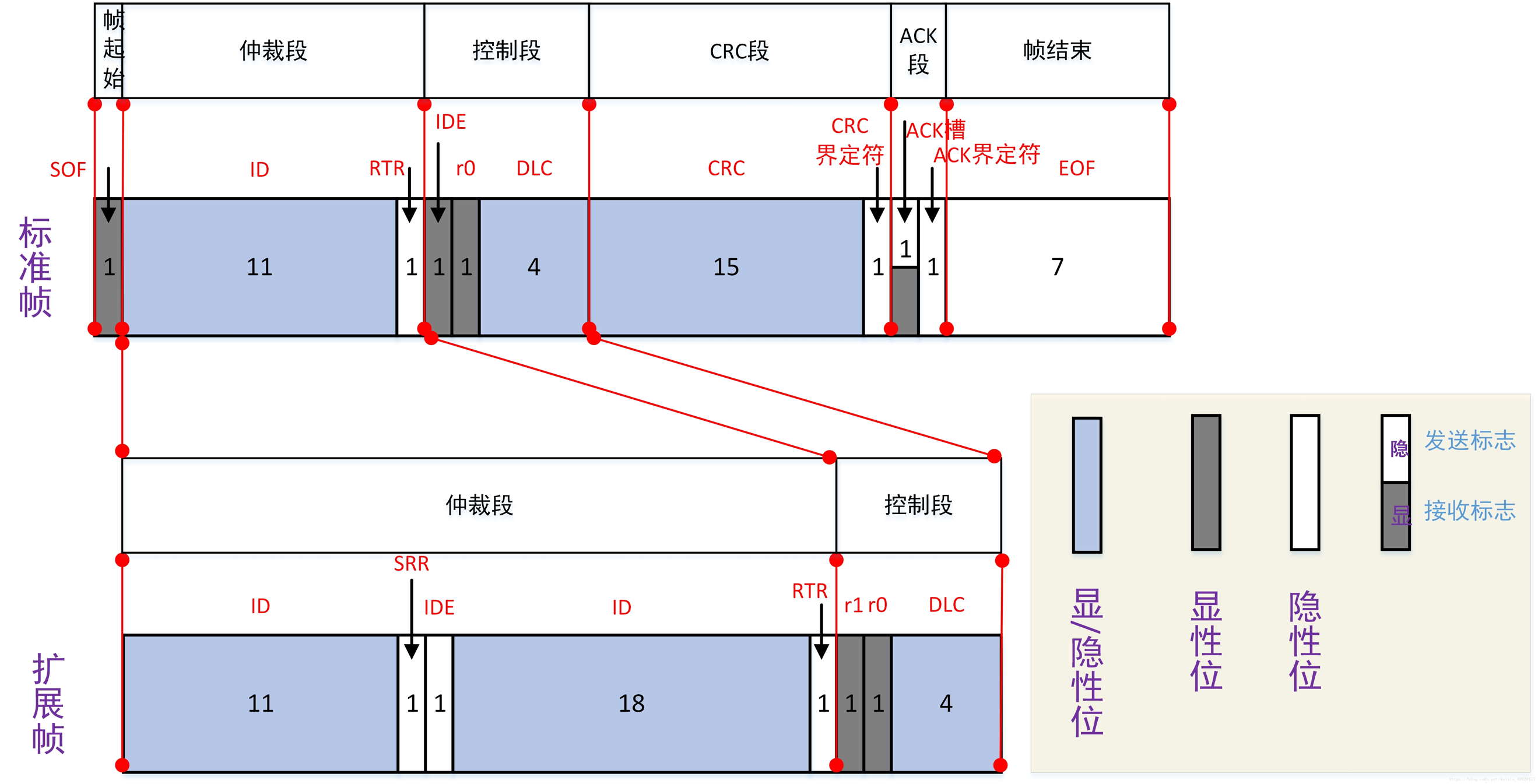
1. 控制段

* r1 : 1位，隐性电平0
* r0 : 1， 0
* DLC: 4位，数据段长度

1. 数据段，crc, ack ,帧结束。同标准帧
2. 远程帧

远程帧分为6个段，也分为标准帧和扩展帧，且RTR位为1（隐性电平）

 较 内 容 
SRR 
是 否 有 数 段 
CRC 校 范 匡 
数 据 帧 
发 送 节 点 的 ID 
0 〈 显 性 电 平 冫 
0 （ 显 性 电 平 ） 
发 送 数 据 长 度 
帧 起 始 十 仲 裁 段 · 
崆 制 段 数 据 縵 
远 程 帧 
被 请 求 发 送 节 点 的 《 D 
1 〈 性 电 平 冫 
1 （ 性 电 平 ） 
谲 求 的 敷 长 度 
帧 始 仲 裁 段 + 控 制 段 

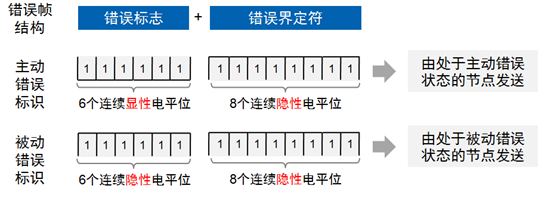


1. 错误帧

五种错误：

* CRC错误：
* 格式错误
* 应答错误：发送节点在ACK阶段没有收到应答信息发生该错误
* 位发送错误
* 位填充错误

当发生上述任何一个错误时，发送错误帧。



1. 过载帧

当某节点没有做好接收的“准备”时，将发送过载帧，以通知发送节点。

 载 帧 经 檢 ： 
6 个 连 性 电 平 位 
过 謔 这 
8 个 连 甲 位 
由 于 存 在 多 个 节 点 同 时 过 且 过 幀 发 送 有 时 间 差 问 0 生 可 出 现 过 栽 标 志 
叠 加 后 超 过 6 个 位 的 觋 象 。 
卜 过 載 吓 忐 删 卜 过 帧 界 定 苻 闷 
卜 一 过 标 志 重 噁 部 分 

1. 帧间隔

错误帧和过载帧前面不加帧间隔。

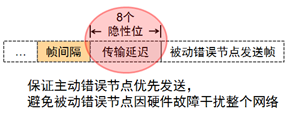
帧间隔用于将数据帧或远程帧和他们之前的帧分离开，但过载帧和错误帧前面不会插入帧间隔。

https://img-blog.csdn.net/20170122155020489

帧间隔过后，如果无节点发送帧，则总线进入空闲。

https://img-blog.csdn.net/20170122155020830

帧间隔过后，如果被动错误节点要发送帧，则先发送8个隐性电平的传输延迟，再发送帧。



# CAN 验证计划

## 验证功能

1. 上电复位 ，配置寄存器，初始化can

* CAN 复位模式
* 设置滤波寄存器
* 波特率
* 输出模式
* 退出复位模式

1. 发送报文
2. 接收报文

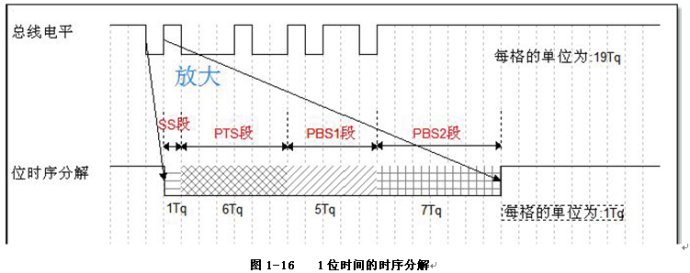
## 波特率计算

CAN异步串行传输。通信双方通过一样的波特率来进行数据采样。

波特率：每秒传输的bit个数。

通过设置BTR0 和BTR1两个寄存器来设置波特率。

为了实现位同步，CAN协议把每一位的时序分解为如下图所示：



* SS段：同步段，如当总线上出现帧起始信号(SOF)时，其它节点上的控制器根据总线上的这个下降沿，对自己的位时序进行调整，把该下降沿包含到SS 段内，这样根据起始帧来进行同步的方式称为硬同步。其中SS 段的大小为1Tq。
* PTS段：

CAN波特率计算：

在CAN底层协议将CAN数据的每一位时间（TBit）分为许多时间段（Tscl）,这些时间段分为：

（1）位同步时间（Tsync）:占一个Tscl

（2）时间段1（Tseg1）：占用（Tseg1＋1） 个Tscl

（3）时间段2（Tseg2）：占用（Tseg2+1）个Tscl

位时间(TBit)就是：

TBit=Tseg1+Tseg2+Tsync=(TSEG1+TSEG2+3)\*Tscl

那么CAN的波特率 (CANbps)就是 ＝ 1/TBit。

波特率范围？

CAN在技术上便引入了重同步的概念，以更好的解决这些问题。这样重同步带来的结果就是要么时间段1(Tseg1)增加TSJW（同步跳转宽度SJW+1），要么时间段减少TSJW，因此CAN的波特率实际上有一个范围：1/(Tbit+Tsjw) ≤CANbps≤1/(Tbit-Tsjw)

CAN有波特率的值四以下几个元素决定：

A． 最小时间段Tscl；

B． 时间段1 TSEG1；

C． 时间段2 TSEG2；

D． 同步跳转宽度 SJW

如何计算Tscl？

总线时序寄存器中的预分频寄存器BRP

Tscl = (BRP + 1)/ Fvbp

Fvbp 是微处理器的外设时钟频率

例子：

假设设定的波特率为CAN bps = 250Khz，总线频率Fvbp ＝ 16M

位时间TBit = 1/ CAN bps = Tscl \*(3+TSEG1+TSEG2)

一般(3+TSEG1+TSEG2)取值范围为（8～25），若取(3+TSEG1+TSEG2)值为16，

得到 Tscl = 250 ns

Tscl = (BRP + 1)/ Fvbp

BRP = 3

根据下表：

2.3.3 MSCAN O (CANBTRO) 
CANBTRO MSCAN CAN 
SJWI 

JWI 
12_4. 
SJWO 
12-5. 
BRP5 BRP4 BRP3 BRP2 BRPI BRPO 

因为计算的BRP为3，所以预分频器值为4。

假设先不考虑SJW位和SAM位。那么，BTR0和BTR1就是2个分频系数寄存器。

BTR0 \* BTR1 = Fvbp/CANbps = 64 = 4 \* BTR1 ,所以BTR1 = 16

|  |  |
| --- | --- |
| 波特率 > 800 K | Sample = 75% |
| 波特率 > 500 K | Sample = 80% |
| 波特率 <= 500 K | Sample = 87.5% |

Sample = [1 + (TSEG1 + 1)] / [1 + (TSEG1 + 1) + (TSEG2 + 1)]

取CAN\_BS1＝TSEG1+1 ＝13，CAN\_BS2 ＝TSEG2+1＝2，sample = (1+13)/(1+13+2)

得到TSEG1 = 12, TSEG2 = 1 ,

SAM的确定：低频时，选SAM＝1，即采样3次。高频100K以上时，取SAM＝0，即采样1次。取SAM = 0,所以BTR1 = 0x1c

注：总体配置保持CAN\_BS1>=CAN\_BS2 ， CAN\_BS1>=1个CAN时钟周期 ，CAN\_BS2>=2SJW

所以SJM = 1, BRT0 = 0x43

## 接收滤波器的设置

总线上数据帧的ID通过待接收节点的验收[滤波](http://www.eepw.com.cn/news/listbylabel/label/%E6%BB%A4%E6%B3%A2)后是吻合的，是可以被接收的。

1. 单滤波
2. 双滤波

双滤波，就是有两次的滤波，但并非两次滤波都需要通过才双通过。

两次滤波只要有一次滤波成功那么就默认滤波通过，可以接收数据了。

CRO 
AMRO 
ID.28-lD.21 
ACRI 
AMRI 
ID.20-lD.13 
ACR2 
AMR2 
ID.2

ACR寄存器是接收代码寄存器，AMR是接收屏蔽寄存器。ACR一般是需要与对应的ID相吻合的，但是如果AMR的相应位上设置为1的时候，ID的那一位数据可以不和ACR的相应位一样，也就是起到屏蔽的作用。

如果ACR0=11101111，AMR0=00000000，那么要想通过验收滤波，必须ID.28-ID.21=ACR0=11101111。如果AMR0=00010000，那么ID.28-ID.21=11111111时，也可以通过验收滤波，因为此时AMR0的第五位为1，也就是屏蔽了ACR0的第五位。所以ID的相应位可以不合ACR0一致。

## 总线空闲与忙

隐性位（1），一旦检测到从隐性电平到显性电平的跳变，就说明总线上有站点开始发送数据。CAN总线空闲的定义是连续11个位的隐性电平（逻辑1），一般用于总线关闭时的自动恢复。CAN的显性电平逻辑0，一旦有显性就说明肯定至少有1个节点发送，那就不是空闲了。CAN的帧间隙是连续3个以上隐性电平。

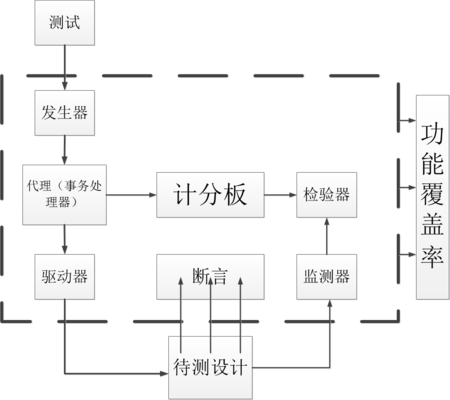
## 同步

## 验证指标

1. 功能覆盖率

对功能覆盖率的测量和使用包括下面几个步骤：  
（1）在测试平台中加入代码，用于监控进入设备中的激励，以及设备对激励的反应，并据此确定哪些功能已经被验证过。  
（2）运行几次仿真，每次使用不同的种子。  
（3）把这些仿真的结果合并到一个报告中。  
（4）对结果进行分析，最后决定如何采用新的激励来达到那些尚未被测试到的条件和逻辑。

1. 。。。



# System Verilog 验证

## 接口

如何抽象出接口？

## 激励

1. 随机激励
2. 定向激励

## 检测器

## 测试报告

信息标准化的打印方法：信息包括“出处”和“内容”。

用库封装报告要求，使用时，配置好信息的出处、种类和重要级别，针对不同的信息种类做出不同的处理方法。

* 信息包括报告出处和内容
* 具备信息类型
* 具备信息重要级别
* 可控的仿真行为和打印通道。

# UVM验证

# 结论