



# (12)发明专利申请

(10)申请公布号 CN 109167737 A  
(43)申请公布日 2019.01.08

(21)申请号 201810844514.3

(22)申请日 2018.07.27

(71)申请人 中国地质大学(武汉)

地址 430000 湖北省武汉市洪山区鲁磨路  
388号

(72)发明人 王宏 韩康 何迪

(74)专利代理机构 武汉知产时代知识产权代理  
有限公司 42238

代理人 邹桂敏

(51) Int. Cl.

H04L 12/863(2013.01)

H04L 12/865(2013.01)

H04L 12/40(2006.01)

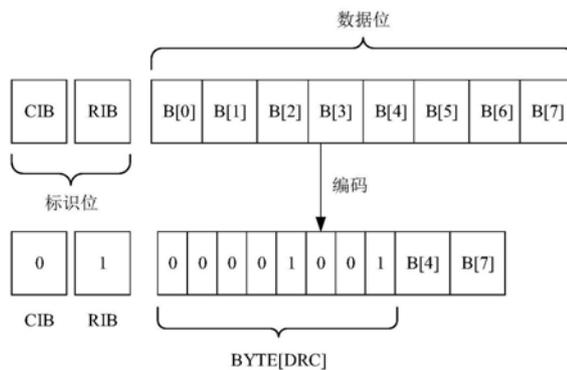
权利要求书2页 说明书6页 附图3页

## (54)发明名称

融合数据缩减与动态优先权队列的CAN调度方法

## (57)摘要

本发明公开了融合数据缩减与动态优先权队列的CAN调度方法,通过对CAN总线协议进行合理地设计,首先通过肯达尔排队理论建立CAN总线消息响应时间与负载的数学模型,得到CAN总线在不同负载下的理论延时;然后针对CAN总线信息重复率高的特点对实时信息进行数据缩减调度,降低CAN总线上的负载,从而提升CAN总线实时性能,并应用分布式队列动态优先级调度算法对CAN总线实时消息进行合理的调度,通过小幅度牺牲高优先级消息实时性能,从而大幅度提高低优先级消息实时性能,从而降低CAN总线消息的平均延迟时间。本发明改善CAN总线系统随着总线长度增加造成的负载能力降低,高负载下实时性能较差等问题。



1. 融合数据缩减与动态优先权队列的CAN调度方法,其特征在于,通过对CAN总线协议进行合理地设计,首先通过肯达尔排队理论建立CAN总线消息响应时间与负载的数学模型,得到CAN总线在不同负载下的理论延时;然后针对CAN总线信息重复率高的特点对实时信息进行数据缩减调度,降低CAN总线上的负载,从而提升CAN总线实时性能,并应用分布式队列动态优先级调度算法对CAN总线实时消息进行合理的调度,通过小幅度牺牲高优先级消息实时性能,从而大幅度提高低优先级消息实时性能,从而降低CAN总线消息的平均延迟时间。

2. 根据权利要求1所述的融合数据缩减与动态优先权队列的CAN调度方法,其特征在于,所述数据缩减调度具体包括以下步骤:

(1-1) 在CAN标识符中取2位,分别用于设置为变化标识位和缩减标识位,变化标识位用以表示数据是否发生变化,缩减标识位用来显示数据重复字节情况;

(1-2) 根据数据重复情况的不同,通过置位不同的变化标识位和缩减标识位来改变消息优先级;

(1-3) 对于进行了数据缩减的消息,即变化标识位和缩减标识位均不为0时,需要对其进行重新编码并产生一个新的字节称之为DRC字节,通过将DRC字节中的第*i*位进行置0来表示当前数据的第*i*个字节与备份数据的第*i*个字节重复,该字节不需发送,故将其缩减;反之,通过将DRC位置1,来表示当前该字节没有重复,并将其按照原有字节先后顺序排列在DRC字节后;

(1-4) 数据经过缩减传输之后,接收节点要对接收信息进行解码重构。

3. 根据权利要求2所述的融合数据缩减与动态优先权队列的CAN调度方法,其特征在于,所述步骤(1-2)中,消息中数据若重复字节数小于等于1字节,则将变化标识位和缩减标识位均置0,其优先级最高;超过1字节数据重复时,则将变化标识位置0,缩减标识位置1,其优先级次之;若数据完全重复,则将变化标识位和缩减标识位均置1,其优先级最低。

4. 根据权利要求2所述的融合数据缩减与动态优先权队列的CAN调度方法,其特征在于,所述步骤(1-4)中,当接收到新消息,接收节点首先判断该数据的变化标识位和缩减标识位,若发现变化标识位置1,则直接调用备份数据;如果变化标识位为0,缩减标识位为1,则接收节点就对该数据进行解码,用DRC段来判断数据的重复情况并调用对应字节的数据,并接收后续第二字节开始的非重复数据,接收节点由此可以复原该消息的全部数据内容。

5. 根据权利要求2所述的融合数据缩减与动态优先权队列的CAN调度方法,其特征在于,所述分布式队列动态优先级调度算法具体包括以下步骤:

(2-1) 取余下CAN标识符用于分布式动态优先队列调度,在余下CAN标识符的所有位ID中取1位作为优先权字段设为 $Q_{pi}$ ,该字段在CAN总线上形成一个FIFO优先权队列,CAN总线上每个节点除了存储了本条消息在队列中的位置,同时需要记录该节点在CAN总线上的等待延时,从而使CAN总线中发送的消息形成一个与其等待时间等参数相关的分布式动态优先权队列;

(2-2) 为不同节点设置不同的晋升权重,高优先级消息应占有较大的晋升权重,低优先级消息的占据较小的晋升权重。

6. 根据权利要求5所述的融合数据缩减与动态优先权队列的CAN调度方法,其特征在于,所述步骤(2-1)中,设节点 $i=0,1,2,\dots,m$ 的优先权字段 $Q_{pi}$ 随消息帧在队列中的等待时间

t动态变化,则其可以表达为如式(1)所示:

$$Q_{pi} = (2^l - 1) - f_i(t) \quad (1)$$

式中: $f_i(t)$ 代表的是随着时间增长而递增的函数,而等待时间主要来源于消息在仲裁过程中竞争不断失败所造成的延时,因此,若设 $\tau$ 为消息帧在CAN总线上的传输时间, $n$ 为消息帧在排队过程中的失败次数,则可进行归一化处理使 $t = n\tau$ ,则可表达为如式(2)所示:

$$Q_{pi} = (2^l - 1) - f_i(n\tau) \quad (2)。$$

7.根据权利要求6所述的融合数据缩减与动态优先权队列的CAN调度方法,其特征在于,所述步骤(2-2)中,选取一部分字段用来做静态优先权字段,余下为动态优先权字段,根据节点固定优先级的不同,为不同的节点来设置了不同的权重,设 $k$ 为优先级晋升权重,各节点根据固定优先级的不同来选取不同的 $k$ 值,固定优先级越高,则 $k$ 越大,反之则越小,由此,式(2)可改为式(3)所示:

$$Q_{pi} = (2^l - 1) - f_i(n, \tau, k) \quad (3)。$$

8.根据权利要求7所述的融合数据缩减与动态优先权队列的CAN调度方法,其特征在于,所述动态优先权晋升队列放在所有CAN标识符的最前面,将变化标识位和缩减标识位放在中间,静态标识符放在最后。

## 融合数据缩减与动态优先权队列的CAN调度方法

### 技术领域

[0001] 本发明涉及工业网络控制技术领域,尤其涉及一种融合数据缩减与动态优先权队列的CAN调度方法。

### 背景技术

[0002] CAN总线中传输的消息根据时间约束要求主要分为强实时消息、弱实时消息和非实时消息。强实时消息主要是指突发性消息,该类消息必须在截止期内传输完毕,为了避免出现CAN总线堵塞和饱和现象,在有限的带宽情况下,需要采取相应的总线消息调度策略,用以合理分配带宽资源,提高CAN消息数据的利用率,提高系统公平性,达到均衡总线负载、降低系统整体延时的目的。为此,需要设计一种CAN总线的消息传输调度方法。

### 发明内容

[0003] 有鉴于此,本发明的实施例提供了一种提高现有CAN总线实时性能的同时也增强其负载能力的融合数据缩减与动态优先权队列的CAN调度方法。

[0004] 本发明的实施例提供融合数据缩减与动态优先权队列的CAN调度方法,通过对CAN总线协议进行合理地设计,首先通过肯达尔排队理论建立CAN总线消息响应时间与负载的数学模型,得到CAN总线在不同负载下的理论延时;然后针对CAN总线信息重复率高的特点对实时信息进行数据缩减调度,降低CAN总线上的负载,从而提升CAN总线实时性能,并应用分布式队列动态优先级调度算法对CAN总线实时消息进行合理的调度,通过小幅度牺牲高优先级消息实时性能,从而大幅度提高低优先级消息实时性能,从而降低CAN总线消息的平均延迟时间。

[0005] 进一步,所述数据缩减调度具体包括以下步骤:

[0006] (1-1) 在CAN标识符中取2位,分别用于设置为变化标识位和缩减标识位,变化标识位用以表示数据是否发生变化,缩减标识位用来显示数据重复字节情况;

[0007] (1-2) 根据数据重复情况的不同,通过置位不同的变化标识位和缩减标识位来改变消息优先级;

[0008] (1-3) 对于进行了数据缩减的消息,即变化标识位和缩减标识位均不为0时,需要对其进行重新编码并产生一个新的字节称之为DRC字节,通过将DRC字节中的第i位进行置0来表示当前数据的第i个字节与备份数据的第i个字节重复,该字节不需发送,故将其缩减;反之,通过将DRC位置1,来表示当前该字节没有重复,并将其按照原有字节先后顺序排列在DRC字节后;

[0009] (1-4) 数据经过缩减传输之后,接收节点要对接收信息进行解码重构。

[0010] 进一步,所述步骤(1-2)中,消息中数据若重复字节数小于等于1字节,则将变化标识位和缩减标识位均置0,其优先级最高;超过1字节数据重复时,则将变化标识位位置0,缩减标识位位置1,其优先级次之;若数据完全重复,则将变化标识位和缩减标识位均置1,其优先级最低。

[0011] 进一步,所述步骤(1-4)中,当接收到新消息,接收节点首先判断该数据的变化标识位和缩减标识位,若发现变化标识位置1,则直接调用备份数据;如果变化标识位为0,缩减标识位为1,则接收节点就对该数据进行解码,用DRC段来判断数据的重复情况并调用对应字节的数据,并接收后续第二字节开始的非重复数据,接收节点由此可以复原该消息的全部数据内容。

[0012] 进一步,所述分布式队列动态优先级调度算法具体包括以下步骤:

[0013] (2-1)取余下CAN标识符用于分布式动态优先队列调度,在余下CAN标识符的所有位ID中取1位作为优先权字段设为 $Q_{pi}$ ,该字段在CAN总线上形成一个FIFO优先权队列,CAN总线上每个节点除了存储了本条消息在队列中的位置,同时需要记录该节点在CAN总线上的等待延时,从而使CAN总线中发送的消息形成一个与其等待时间等参数相关的分布式动态优先权队列;

[0014] (2-2)为不同节点设置不同的晋升权重,高优先级消息应占有较大的晋升权重,低优先级消息的占据较小的晋升权重。

[0015] 进一步,所述步骤(2-1)中,设节点 $i=0,1,2\cdots,m$ 的优先权字段 $Q_{pi}$ 随消息帧在队列中的等待时间 $t$ 动态变化,则其可以表达为如式(1)所示:

$$[0016] \quad Q_{pi} = (2^1 - 1) - f_i(t) \quad (1)$$

[0017] 式中: $f_i(t)$ 代表的是随着时间增长而递增的函数,而等待时间主要来源于消息在仲裁过程中竞争不断失败所造成的延时,因此,若设 $\tau$ 为消息帧在CAN总线上的传输时间, $n$ 为消息帧在排队过程中的失败次数,则可进行归一化处理使 $t=n\tau$ ,则可表达为如式(2)所示:

$$[0018] \quad Q_{pi} = (2^1 - 1) - f_i(n\tau) \quad (2)。$$

[0019] 进一步,所述步骤(2-2)中,选取一部分字段用来做静态优先权字段,余下为动态优先权字段,根据节点固定优先级的不同,为不同的节点来设置了不同的权重,设 $k$ 为优先级晋升权重,各节点根据固定优先级的不同来选取不同的 $k$ 值,固定优先级越高,则 $k$ 越大,反之则越小,由此,式(2)可改为式(3)所示:

$$[0020] \quad Q_{pi} = (2^1 - 1) - f_i(n, \tau, k) \quad (3)。$$

[0021] 进一步,所述动态优先权晋升队列放在所有CAN标识符的最前面,将变化标识位和缩减标识位放在中间,静态标识符放在最后。

[0022] 与现有技术相比,本发明具有以下有益效果:针对一部分CAN总线系统数据重复率高的特点,提出了一种数据缩减调度算法(Data Reduction,DR)来改善总线负载情况,降低了总线的负载率从而提升了系统的实时性能和负载能力。针对高负载情况下CAN总线实时性能较差的情况,使用了分布式队列动态优先级调度算法(Distributed Dynamic Priority Queue,DDPQ)来改善系统高负载情况下的公平性,进而提升系统总体实时性能。结合DR算法和DDPQ算法各自优点,通过合理地设计协议,DDPQDR算法在负载能力和实时性能上的表现更为优异,从而来改善CAN总线系统随着总线长度增加造成的负载能力降低,高负载下实时性能较差等问题。

## 附图说明

[0023] 图1是本发明一实施例数据缩减调度中的算法编码原理图。

- [0024] 图2是本发明一实施例数据缩减调度中的解码流程图。
- [0025] 图3是本发明一实施例分布式队列动态优先级调度算法晋升流程图。
- [0026] 图4是本发明一实施例具体标识符字段示意图。
- [0027] 图5是本发明一实施例负载和延时分析图。

### 具体实施方式

[0028] 为使本发明的目的、技术方案和优点更加清楚,下面将结合附图对本发明实施方式作进一步地描述。

[0029] 本发明的实施例提供了融合数据缩减与动态优先权队列的CAN调度方法,通过对CAN总线协议进行合理地设计,首先通过肯达尔排队理论建立CAN总线消息响应时间与负载的数学模型,得到CAN总线在不同负载下的理论延时;然后针对CAN总线信息重复率高的特点对实时信息进行数据缩减调度,降低CAN总线上的负载,从而提升CAN总线实时性能,并应用分布式队列动态优先级调度算法对CAN总线实时消息进行合理的调度,通过小幅度牺牲高优先级消息实时性能,从而大幅度提高低优先级消息实时性能,从而降低CAN总线消息的平均延迟时间。

[0030] 数据缩减调度具体包括以下步骤:

[0031] (1-1) 在CAN标识符中取2位,分别用于设置为变化标识位和缩减标识位,变化标识位用以表示数据是否发生变化,缩减标识位用来显示数据重复字节情况;

[0032] (1-2) 根据数据重复情况的不同,通过置位不同的变化标识位和缩减标识位来改变消息优先级;消息中数据若重复字节数小于等于1字节,则将变化标识位和缩减标识位均置0,其优先级最高;超过1字节数据重复时,则将变化标识位置0,缩减标识位置1,其优先级次之;若数据完全重复,则将变化标识位和缩减标识位均置1,其优先级最低;

[0033] (1-3) 对于进行了数据缩减的消息,即变化标识位和缩减标识位均不为0时,需要对其进行重新编码并产生一个新的字节称之为DRC字节,通过将DRC字节中的第i位进行置0来表示当前数据的第i个字节与备份数据的第i个字节重复,该字节不需发送,故将其缩减;反之,通过将DRC位置1,来表示当前该字节没有重复,并将其按照原有字节先后顺序排列在DRC字节后;

[0034] (1-4) 数据经过缩减传输之后,接收节点要对接收信息进行解码重构;当接收到新消息,接收节点首先判断该数据的变化标识位和缩减标识位,若发现变化标识位置1,则直接调用备份数据;如果变化标识位为0,缩减标识位为1,则接收节点就对该数据进行解码,用DRC段来判断数据的重复情况并调用对应字节的数据,并接收后续第二字节开始的非重复数据,接收节点由此可以复原该消息的全部数据内容。

[0035] 分布式队列动态优先级调度算法具体包括以下步骤:

[0036] (2-1) 取余下CAN标识符用于分布式动态优先队列调度,在余下CAN标识符的所有位ID中取1位作为优先权字段设为 $Q_{pi}$ ,该字段在CAN总线上形成一个FIFO优先权队列,CAN总线上每个节点除了存储了本条消息在队列中的位置,同时需要记录该节点在CAN总线上的等待延时,从而使CAN总线中发送的消息形成一个与其等待时间等参数相关的分布式动态优先权队列;

[0037] 设节点 $i=0,1,2,\dots,m$ 的优先权字段 $Q_{pi}$ 随消息帧在队列中的等待时间 $t$ 动态变化,

则其可以表达为如式(1)所示:

$$[0038] \quad Q_{pi} = (2^1 - 1) - f_i(t) \quad (1)$$

[0039] 式中: $f_i(t)$ 代表的是随着时间增长而递增的函数,而等待时间主要来源于消息在仲裁过程中竞争不断失败所造成的延时,因此,若设 $\tau$ 为消息帧在CAN总线上的传输时间, $n$ 为消息帧在排队过程中的失败次数,则可进行归一化处理使 $t = n\tau$ ,则可表达为如式(2)所示:

$$[0040] \quad Q_{pi} = (2^1 - 1) - f_i(n\tau) \quad (2)。$$

[0041] (2-2)为不同节点设置不同的晋升权重,高优先级消息应占有较大的晋升权重,低优先级消息的占据较小的晋升权重。

[0042] 选取一部分字段用来做静态优先权字段,余下为动态优先权字段,根据节点固定优先级的不同,为不同的节点来设置了不同的权重,设 $k$ 为优先级晋升权重,各节点根据固定优先级的不同来选取不同的 $k$ 值,固定优先级越高,则 $k$ 越大,反之则越小,由此,式(2)可改为式(3)所示:

$$[0043] \quad Q_{pi} = (2^1 - 1) - f_i(n, \tau, k) \quad (3)。$$

[0044] 动态优先权晋升队列放在所有CAN标识符的最前面,将变化标识位和缩减标识位放在中间,静态标识符放在最后。

[0045] 实施例1

[0046] 本发明的实施例提供了融合数据缩减与动态优先权队列的CAN调度方法,包括:

[0047] (1)针对一部分CAN总线系统数据重复率高的特点,提出了一种数据缩减调度方法(Data Reduction,DR)来改善总线负载情况,从而来提升系统的实时性能和负载能力。

[0048] (2)针对高负载情况下CAN总线实时性能较差的情况,使用了分布式队列动态优先级调度算法(Distributed Dynamic Priority Queue,DDPQ)来改善系统高负载情况下的公平性,进而提升系统总体实时性能。

[0049] (3)为了结合DR算法和DDPQ算法各自优点,本发明提出了一种融合数据缩减的分布式队列优先级的改良调度算法(DDPQDR)。通过合理地设计协议,DDPQDR算法在负载能力和实时性能上的表现更为优异。

[0050] 数据缩减调度包括以下子步骤:

[0051] (1-1)以CAN2.0A标准为例,取11位CAN标识符中的2位用于设置为变化标识位(Change Identification Bit,CIB)和缩减标识位(Reduced Identification Bit,RIB)。CIB位用以表示数据是否发生变化,RIB位用来显示数据重复字节情况。并根据数据重复情况的不同,通过置位CIB和RIB来改变消息优先级。

[0052] (1-2)根据重复情况的不同,设置不同的CIB位和RIB位。消息中数据若重复字节数小于等于1字节,则将CIB位和RIB位均置0,其优先级最高;超过1字节数据重复时,则将CIB置0,RIB置1,其优先级次之。若数据完全重复,则将CIB和RIB均置1,其优先级最低。CIB位和RIB位标识符具体变化如表1所示。

[0053] 表1 缩减标识位变换表

	数据重复情况	CIB	RIB
	无重复	0	0
[0054]	仅1字节重复	0	0
	重复字节>1	0	1
	完全重复	1	1

[0055] (1-3) 对于进行了数据缩减的消息,即CIB和RIB均不为0时,需要对其进行重新编码并产生一个新的字节称之为DRC字节,通过将DRC字节中的第*i*位进行置0来表示当前数据的第*i*个字节与备份数据的第*i*个字节重复,因此该字节不需发送,故将其缩减。反之通过将DRC位置1,来表示当前该字节没有重复,并将其按照原有字节先后顺序排列在DRC字节后。假设下面一消息仅有第4号和第7号字节发生变化,其余全是重复字节,其编码原理如图1所示。

[0056] (1-4) 数据经过缩减传输之后,接收节点要对接收信息进行解码重构,如图2所示。当接收到新消息,接收节点首先判断该数据的CIB位与RIB位。若发现CIB置1,则直接调用备份数据。如果CIB为0,RIB为1,则接收节点就对该数据进行解码,用DRC段来判断数据的重复情况并调用对应字节的数据,并接收后续第二字节开始的非重复数据,接收节点由此可以复原该消息的全部数据内容。

[0057] 分布式队列动态优先级调度算法包括以下子步骤:

[0058] (2-1) 在CAN标识符的11位ID中取1位作为优先权字段设为 $Q_{pi}$ ,该字段在总线上形成一个FIFO优先权队列,总线上每个节点除了存储了本条消息在队列中的位置,同时需要记录该节点在总线上的等待延时,从而使总线中发送的消息形成一个与其等待时间等参数相关的分布式动态优先权队列。设节点*i*=0,1,2...*m*的优先权字段 $Q_{pi}$ 随消息帧在队列中的等待时间*t*动态变化,则其可以表达为如式(1)所示。

$$[0059] \quad Q_{pi} = (2^1 - 1) - f_i(t) \quad (1)$$

[0060]  $f_i(t)$  代表的是随着时间增长而递增的函数。而等待时间主要来源于消息在仲裁过程中竞争不断失败所造成的延时,因此,若设 $\tau$ 为消息帧在总线上的传输时间,*n*为消息帧在排队过程中的失败次数,则可进行归一化处理使 $t = n\tau$ ,则可表达为如式(2)所示:

$$[0061] \quad Q_{pi} = (2^1 - 1) - f_i(n\tau) \quad (2)$$

[0062] (2-2) 为不同节点设置不同的晋升权重,高优先级消息应占有较大的晋升权重,低优先级消息的占据较小的晋升权重来解决过分强调公平性的问题。本发明选取了一部分字段用来做静态优先权字段,根据节点固定优先级的不同,为不同的节点来设置了不同的权重。设*k*为优先级晋升权重,各节点根据固定优先级的不同来选取不同的*k*值。固定优先级越高,则*k*越大,反之则越小。由此,式(2)可改为式(3)所示:

$$[0063] \quad Q_{pi} = (2^1 - 1) - f_i(n, \tau, k) \quad (3)$$

[0064] 这样不仅能降低动态优先级提升过程中调度算法所需大量的系统资源,同时保障各类节点消息的实时性要求。DDPQ算法晋升流程图如图3所示。

[0065] 融合数据缩减的分布式队列优先级的改良调度算法包括以下子步骤:

[0066] (3-1) 通过合理地协议设计,将步骤(1)(2)进行算法上的融合。取7位标识符用于分布式动态优先队列调度;取2位标识符作为静态优先级段;同时占用2位标识符的RIB和

CIB位。像这样分配其各自所占字段,正好占据完11位标识符。

[0067] (3-2)将7位动态优先权晋升队列放在标识符的最前面。将2位缩减标识符放在中间,适当的降低重复数据的优先级。而3位静态标识符放在最后,用来实现节点分类。具体标识符字段示意图如图4所示。

[0068] 其各自的数据编码流程、动态优先权提升过程同步骤(1)与步骤(2)中方法的原理相同,只是各自标识符字节位置发生变化。

[0069] 如图3中,首先对于CIB和RIB位,检测数据重复情况,根据表1给RIB位和CIB位赋值,并进行编解码传输。然后对于动态标识符字段,将标识符中前7位设置为动态优先权,由于CAN总线规定标识符上不能出现连续的7个隐性位,所以优先权最大队列值只能取为126,所以动态优先权队列的取值范围为(0-126)。对于静态标识符字段,利用2位静态标识符设置三种静态优先级,分别为00,01,10,分别从高到低排列赋予其晋升权重取为15:3:1,从而可以得到总线上动态字段晋升权值函数如式(4)所示:

$$[0070] \quad \begin{cases} k = 5x^2 - 17x + 15 \\ f_i(n, \tau, k) = k \times n \end{cases} \quad (4)$$

[0071] 式中,x是优先级类,分别对应节点的0,1,2。从而动态优先权晋升函数可以写为式(5)所示:

$$[0072] \quad Q_{pi} = 126 - (5x^2 - 17x + 15) \times n \quad (5)$$

[0073] 消息帧每次在发送前首先获取本节点的静态优先级x和在排队过程中发生冲突的次数n,当n=0时就是新产生的帧刚刚进入优先权队列中排队的时刻,总线经过仲裁之后将动态优先级最小的消息从队列中发送出去。节点如果在数据发送过程中检测到有多个消息竞争,则按照原有CAN协议仲裁,仲裁失败则将冲突次数加1,然后通过调用 $Q_{pi}$ 提升其动态优先级,计算结束后更新消息动态优先级队列,然后等待下一轮总线仲裁。当动态优先级计算结果小于或等于0,优先级均取0。若标识符动态优先级相同时,则继续仲裁其RIB位和CIB位,根据重复情况的不同来进行调度。

[0074] 在不冲突的情况下,本文中上述实施例及实施例中的特征可以相互结合。

[0075] 以上所述仅为本发明的较佳实施例,并不用以限制本发明,凡在本发明的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

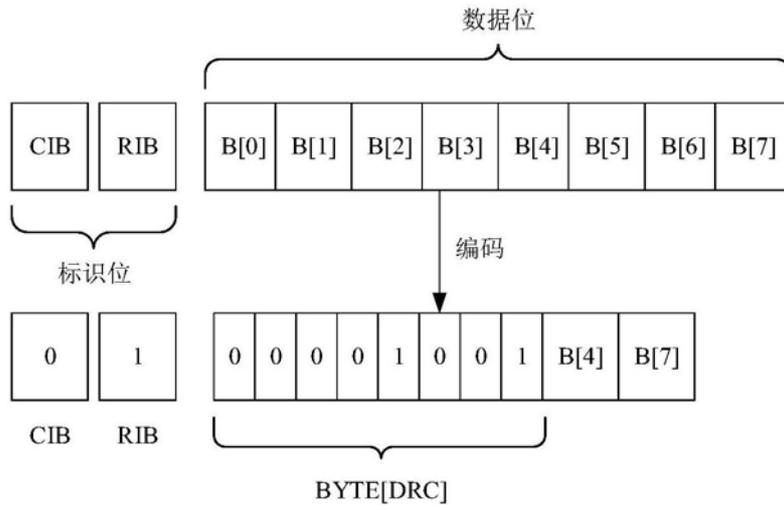


图1

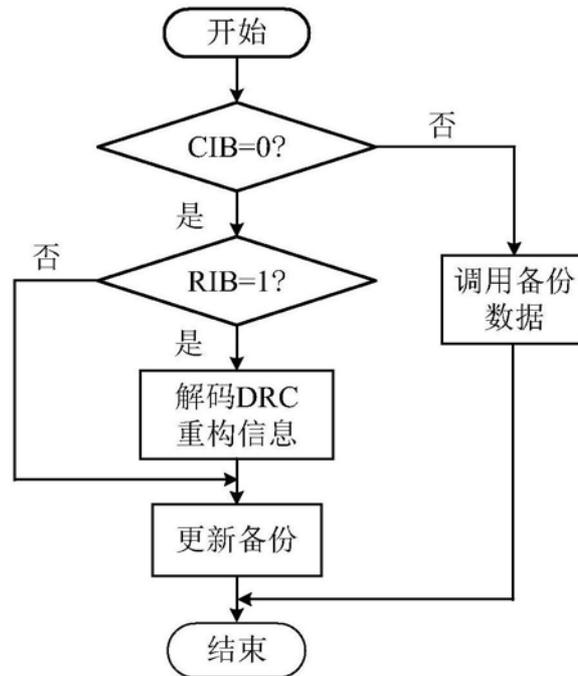


图2

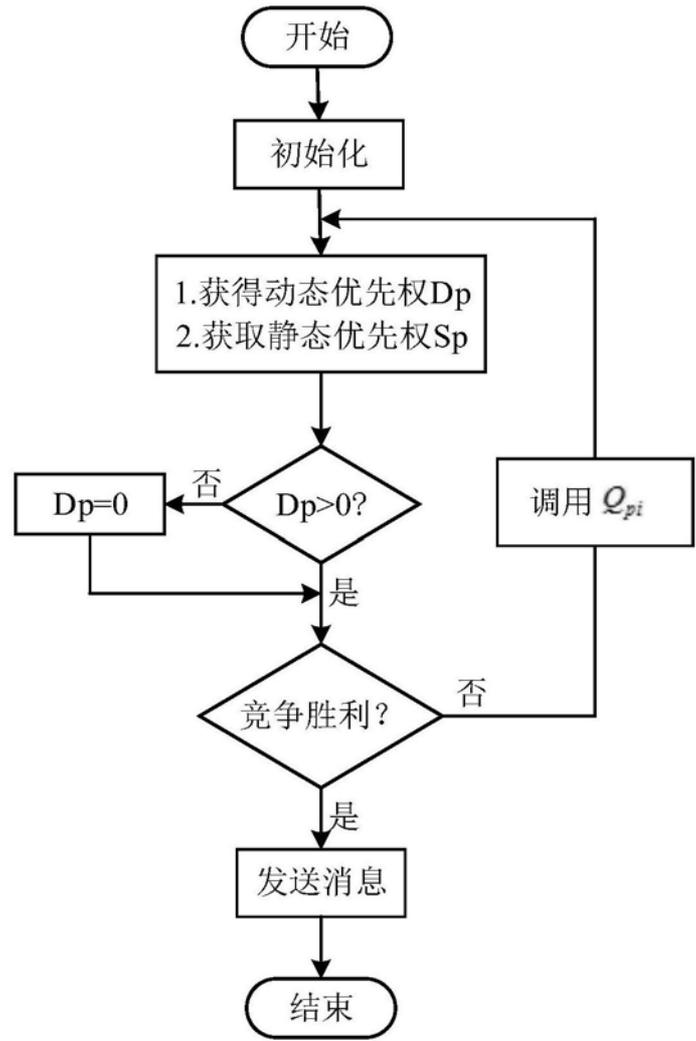


图3

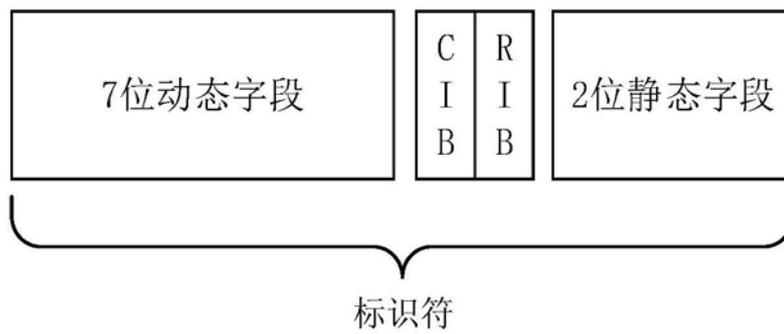


图4

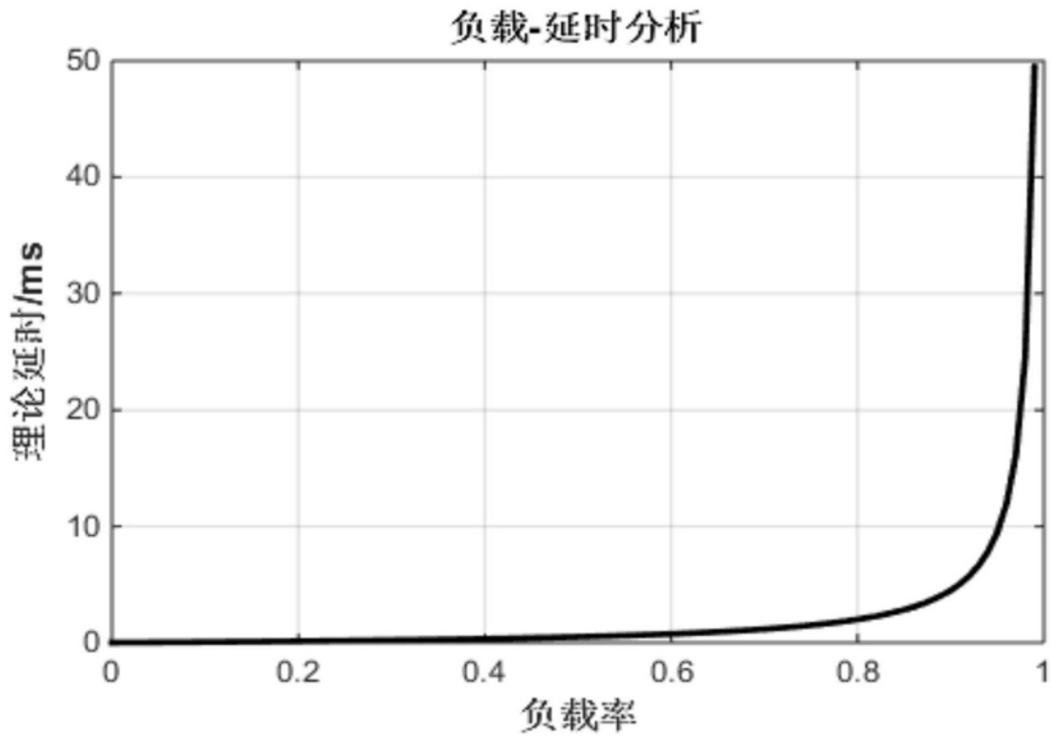


图5