

文章编号: 1002-0411(2004)01-0036-05

Device Net 现场总线的实时性分析及改进

方晓柯¹, 孙盛骐², 王建辉¹, 顾树生¹

(1. 东北大学信息科学与工程学院, 辽宁 沈阳 110004; 2. 沈阳大学信息工程学院, 辽宁 沈阳 110044)

摘要: Device Net 是一种低成本、高性能、应用广泛的设备级现场总线。本文对它的实时性进行了深入分析, 推导出位时间、传输时延和采样点等参数之间的对应关系。在此基础上对其缺点进行了改进, 增加了跳变沿, 并与数据压缩技术相结合, 提高了总线的总体性能。仿真结果表明了这种方法的有效性。*

关键词: Device Net; MAC; 实时性; 现场总线

中图分类号: TP13 **文献标识码:** B

Analysis and Improvement of Real-time Performance for DeviceNet Fieldbus

Fang Xiao-ke¹, Sun Sheng-qi², Wang Jian-hui¹, Gu Shu-sheng¹

(1. The Institute of Information Science and Engineering, Northeastern University, Shenyang 110004, China;

2. The Institute of Information Engineering, Shenyang University, Shenyang 110044, China)

Abstract: The Device Net is a low-price and high-performance fieldbus. It has been applied in lots of fields. The real-time performance of Device Net has been analyzed deeply. Based on the analysis, we have deduced the relation of bit time, transmission delay and sample time. By means of voltage jump and data compression, the performance of the network has been increased. Simulation results show the effectiveness of the proposed method.

Keywords: Device Net; MAC; real-time performance; fieldbus

1 引言 (Introduction)

Device Net(设备网)是美国罗克韦尔自动化公司(Rockwell Automation)于1994年推出的建立在CAN总线基础上的一种低成本、高性能的设备级现场总线^[1]。它是一个开放式的网络标准,其规范和协议完全公开,有众多制造商支持,在全球得到了广泛的推广和应用。它具有可靠性高、实时性好、灵活性强等优点,最适合于对需要多主方式、主要传输突发性数据的系统。但是它也有其缺点:首先,它的传输速率不高,网络扩展能力不好。它只能支持三种波特率:125kbp(500m),250kbp(250m),500kbp(100m)。其次,它采用NRZ(non-return-to-zero,不归零)式数字信号编码,位同步不好,位偏移较大,需采用位填充技术,从而降低了数据传输效率,不能实现变压器隔离,不利于保护通信设备。另外,当节点数目过多、数据发送过于频繁时,网络的传输性能下降。本文主要对Device Net的实时性,即MAC层控制方式,进行深入的分析,并在此基础上对它的缺点

进行了相应的改进。

2 实时性分析 (Real-time performance analysis)

2.1 MAC层控制机制

Device Net 现场总线 MAC 层采用了新型的“非破坏性逐位仲裁”(non-destructive bitwise arbitration)机制对总线的使用权进行控制。这种方法相当于带优先级的CSMA/CD(载波监听多路访问/冲突检测)^[2]。采用CSMA/CD方式时,节点的发送是随机的。各个节点不分主从,先监听总线状态,一旦总线空闲就可以发送数据,若出现碰撞则立即停止,等待一段随机时间后再重复上述过程。这种方法适合于网络负载较轻、节点的发送为突发性的场合。当负载增加时,碰撞几率增加,等待时间的随机性使得数据发送的实时性得不到保证,因此,Device Net 现场总线对其进行了改进,在发送的数据前增加了11位标识符,通过它们来确定数据的优先级。

Device Net 规定了两种总线状态,“显性”(逻辑

* 收稿日期:2003-04-10
基金项目:国家自然科学基金资助项目(60274024)

0) 和“隐性”(逻辑 1). 当同时向总线发送显性位和隐性位时, 总线上出现的是显性位 0. 这样, 当多个节点同时发送数据时, 首先进入 11 位标识符即仲裁区. 在此期间, 每一个发送器都监视总线上当前的电平, 并与它发送的位电平进行比较, 如果值相等, 这个节点继续发送, 如果发送一个隐性位(1), 而在总线上检测到一个显性位(0), 那么此节点失去仲裁

权, 立即停止下一位的发送. 11 位仲裁区值最小的(即优先级最高) 将赢得总线使用权, 其数据不受影响完整地发送出去. Device Net 协议保证每次只有一个节点能够赢得仲裁, 其它节点必须等到此次传输结束后, 总线空闲时再尝试下一次的发送^[3]. 通过这种方式达到了“非破坏, 逐位仲裁”的目的, 实时性得到了保证, 见图 1.



图 1 非破坏性逐位仲裁机制实例

Fig.1 Example of non-destructive bit wise arbitration

2.2 实时性分析

总线上各个节点在发送数据前都要先监听总线状态, 当总线空闲时才会发送数据. 既然如此, 为什么还会产生碰撞呢? 其原因实质上是由于总线时延造成的. 现场总线上的数据是通过物理介质进行传输的, 这必然会存在传输上的时间延迟, 距离越长, 时延越大, 这是构成总线时延的主要原因. 此外, 还有一些其它方面的因素, 如执行总线管理功能的各种器件存在着一定的内部时延等等, 这些都造成了节点对总线状态判断上的失误. 下面我们就深入研究一下冲突发生时的情形. 假设网络上有两个节点 A 和 B, 它们之间有一定的距离. 节点 A 首先发送数据, 数据帧沿着总线传播, 在它的帧头传播到节点 B

前, 节点 B 认为总线处于空闲状态, 所以它也发送了数据, 这样冲突就产生了, 由此进入仲裁阶段.

Device Net 采用了 NRZ(non-return to zero, 不归零) 式数字信号编码, 具有显性和隐性两种状态, 并规定显性为逻辑 0, 隐性为逻辑 1, 实行“线与”操作: 当同时发送显性位和隐性位时, 总线呈现为显性状态. 由于节点 A 先发送和节点 B 先发送的分析方法相同, 所以这里我们假设节点 A 先发数据, 起始时刻为 t_0 . 设节点 A 和节点 B 之间的时延为 τ_{AB} , 每一位持续的时间为 t_{bit} , 节点 A 和节点 B 前三位分别为 011 和 010, 其中第一位是起始位. 仲裁时前三位的情况如图 2 所示.

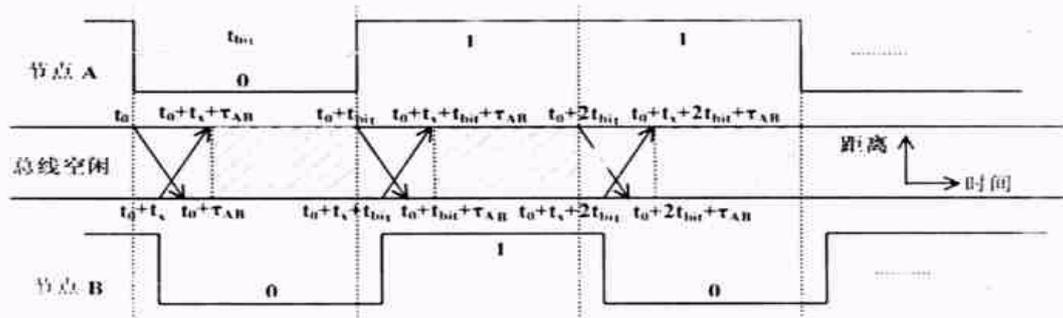


图 2 仲裁过程

Fig.2 Arbitration process

图 2 中横坐标表示时间,纵坐标表示节点沿着总线传播的距离。

当节点 A 检测到总线空闲时于 t_0 时刻开始发送数据,驱动总线为显性 0。由于受到传播延时的影响,该信号在 $t_0 + \tau_{AB}$ 时刻到达 B 节点。在此之前,节点 B 也认为总线是空闲的,并在 $t_0 + t_x$ 时刻开始发送数据,由于它的驱动方向与节点 A 相同,所以总线状态仍为 0。节点 A 在 $t_0 + t_{bit}$ 时刻产生正跳变,想使总线呈现隐性 1 状态,但是节点 B 此时还未开始发送第二位,仍为第一位“0”,经过“线与”操作,总线保持显性 0 状态。节点 A 若在此期间进行仲裁操作必然会得到错误的结论,认为有优先级比自己高的节点在发送数据,自己应退出。因此可以看到,读取总线状态,即采样点的选择,极为重要。

节点 B 在 $t_0 + t_x + t_{bit}$ 时刻产生正跳变,驱动总线为 1,该信号在 $t_0 + t_x + t_{bit} + \tau_{AB}$ 时刻到达 A 节点,总线呈隐性 1 状态。因此节点 A 应该在 $t_0 + t_x + t_{bit} + \tau_{AB}$ 时刻以后进行采样,发现总线电平与自己发送的位电平相同,继续发送。节点 A 第三位为 1,对总线状态无影响,直到节点 B 产生的负跳变在 $t_0 + t_x + 2t_{bit} + \tau_{AB}$ 时刻到达 A 节点时,总线呈现显性状态 0,此时若节点 A 进行仲裁操作,就会发现自己优先级低,自动退出竞争,结束仲裁过程,这时节点 B 不受影响继续发送。设每一位采样点时刻为 at_{bit} ,则有:

$$t_x + \tau_{AB} < at_{bit} < t_{bit} \quad (1)$$

见图 2 中阴影部分。同理,从节点 B 角度来看,正确的采样点时刻应该为:

$$\tau_{AB} < at_{bit} < t_{bit} + t_x \quad (2)$$

并且,设总线中最大的点到点传播延时为 τ_{max} ,得:

$$\tau_{AB} \leq \tau_{max} \quad (3)$$

为了使其具有一般性,取 $t_x = \tau_{AB} = \tau_{max}$ 。令每个节点采样点的选取规则相同,取(1)式和(2)式的交集,最后整理得:

$$2\tau_{max} < at_{bit} < t_{bit} \quad (4)$$

2.3 分析结果

由(4)式,我们可以得出如下结论:

1) Device Net 为低速现场总线。

2) τ_{max} 反映了总线中最大点到点间传播的往返延时,它与总线的物理特性有关,如介质的类型、总线的长度等,当它们选定后,可以认为是常数。 t_{bit} 与总线的波特率 V_{Baud} 成反比,即 $t_{bit} = 1/V_{Baud}$ 。为满

足(4)式的要求, t_{bit} 应足够长,以保证节点能够正常地进行逐位仲裁,因此总线的波特率不会太高,这也是 Device Net 为低速现场总线的本质原因。

2) Device Net 现场总线长度受限,波特率与长度相互制约。

若 Device Net 总线长度增加, $2\tau_{max}$ 值随之变大,则 t_{bit} 也随之加大,波特率降低;反之,为了保证总线具有较高的波特率,则总线长度受限。这是 Device Net 现场总线扩展性差的原因之一。

3) Device Net 现场总线位电平的采样点一般在位时间的 50%~80% 之间。

若将(4)式稍加变换得:

$$a > 2\tau_{max}/t_{bit} \text{ 且 } 0 < a < 1 \quad (5)$$

由(5)式可以看出,虽然从提高总线波特率角度来讲,采样点越靠后越好,即 a 越大, $2\tau_{max}$ 不变时, t_{bit} 可以减小,但是在实际应用时,应为节点读取数值后进行处理保留一定的时间,因此 a 不可以取值太大;反之, a 也不可以取值太小,一方面会使波特率下降,另一方面它应为传播时延。各个节点的同步等留出足够的时间,因此,在实际应用时, a 一般取值为 50%~80%。

3 改进方案(Improvement method)

3.1 物理层的改进

通过上面的分析,我们可对 Device Net 现场总线的物理层进行如下改进:

既然采样点可以位于每一位的后半部分,那么我们可以在前半部分加入跳变沿,设跳变位于 bt_{bit} 处,显性位“0”对应高到低跳变,隐性位“1”对应低到高跳变,采样点时刻仍为 at_{bit} 。同样,采用上述方法进行分析得到:

$$bt_{bit} < t_x < \tau_{AB} \leq \tau_{max} \quad (6)$$

从(5)式和(6)式整理得:

$$b < a/2 \text{ 且 } 0 < a < 1 \quad (7)$$

对于低速总线,这一条件容易满足,例如,设跳变出现在每位的 0~10% 之间,采样点位于每位的 80% 处即可。

由于每一位都有一个跳变,可以实现位同步,而且由于直流分量不会积累,可以采用变压器隔离技术,保证了通讯设备的安全。另外,由于跳变,位偏移小,不用再进行位填充,提高了数据传输效率。并且这一改进并不影响 Device Net 的上层协议,有很好的兼容性。

3.2 减少网络数据流量

从上面的分析中我们可以看到,“非破坏性逐位仲裁”机制是 Device Net 现场总线最具有特色的地方,它解决了 CSMA/CD 方式下实时性得不到保证的缺点,但是与此同时,它也是导致传输速率无法提高的根本原因,影响了网络的传输能力.另一方面,如果节点很多,每个节点发送数据的次数比较频繁,则碰撞几率增加,仲裁次数增多,等待时间延长(最大仲裁时间每次为 $12t_{bit}$),对于优先级低的节点更是如此,传输效率进一步受到影响.这些被传输的数据实际上有很大一部分是原始测量数据,它们通常都会包含大量的冗余和不相关的信息,可以将它们先进行处理,如特征提取、无用数据剔除等,然后再进行传送.这样既可以减少网络上传输的数据流量,又可以控制传输次数,使网络的性能得到改善.

计算机网络、多媒体通信等领域也存在着同样的问题.它们通过数据压缩技术减少网络上的数据流量,提高了网络的整体性能.这种技术已广泛应用于声音、图像信息以及地震、气象等数据的传输中.但是,这些数据压缩方法不能直接用于工业现场,其主要原因是在工业控制中数据压缩和数据恢复都要求有很高的实时性,因此,在工业现场中一般都采用算法简单、运算速度快的方法.

目前,在工业中常用的数据压缩方法主要分为三种^[4]:分段线性法、矢量化方法和信号变换法.其中,矢量化方法由于计算量大,运行速度比较慢,不太适用于现场总线场合.而信号变换法可以实现较大的压缩比,如小波变换等,但是它现在还有许多实际问题需要解决.至于分段线性法,它可以细分为矩

形波串法(Box Car)、后向斜率法(Backward Slope)、转门趋势化法(SDT, Swing Door Trending)、分段线性在线趋势化法(PLOT, Piecewise Linear Online Trending)等.其中 SDT 法在工业中应用较多,尽管它的压缩比不是很高,但是它的突出优点是算法简单,运行速度快.

鉴于 Device Net 现场总线一般用于现场级设备,节点 CPU 处理能力较低,时钟主频不高,运算量不宜太大,所以我们采用一种新型矩形波串法^[5]对数据进行压缩.善了传统的矩形波串方法对数据过程的线性漂移或数据突变过于敏感的缺点,并且算法简单,软件编程一般只需十几条语句,运行速度快,实时性好.至于数据解压,方法比较简单,只要进行线性插值即可.

由于在 Device Net 中 RTR 位目前还预留未用,因此我们可以利用这一位来标识帧的压缩状态,可以设 RTR 位等于 1 时表示此帧为压缩数据, RTR 位等于 0 时表示此帧为未压缩数据.

用户将事先编好的在线压缩程序下载到节点中,并在组态时对新型矩形波串法中的记录限等相关参数进行设定.当节点采集数据时,将原始数据放入缓冲区,然后调用压缩程序,并用信号滤波公式对原始数据进行滤波,经过滤波后的数据我们称之为滤波数据.系统提取滤波数据,并与上一次发出的数据进行比较,若差值大于或等于用户事先指定的记录限则发送该数据,并将帧的 RTR 位置 1;若差值低于指定的记录限,则不发送该数据.工作过程如图 3 所示.

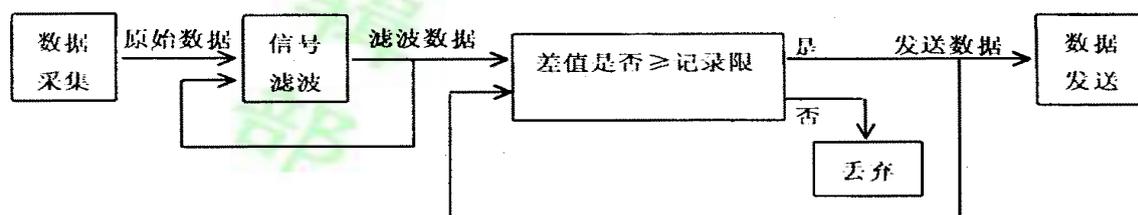


图 3 数据压缩过程示意图

Fig. 3 Schematic diagram of data compression

滤波数据不管最后发出与否,在参与了下一次的信号滤波,产生了新的滤波数据后,这个数据即可丢弃,并由新的滤波数据替代.而发出的数据在发送成功后,仍驻留在缓冲区中,直到参与差值计算并生成新的发出数据后再做丢弃处理.

当节点接收数据时,若发现 $RTR=1$,则得知所接收的是压缩数据,调用相应的压缩程序对其进行

解压,然后再用;若发现 $RTR=0$,则按正常数据进行处理.

从上述过程中可以看到,这种方法对现在的 Device Net 协议改动不大,数据帧结构没有变,只是利用了其未使用的 RTR 位对压缩与否进行标注,但是却可以提高总线的使用效率,在保证兼容性的同时,实现了对网络中数据流量的控制,提高了系统的

总体性能。

4 仿真结果 (Simulation result)

我们以数据在网络传输过程中所需的时间为参

考指标,对压缩前后系统的实时性能进行了对比,其仿真结果如图 4 所示。

图 4 中压缩比为 1 时表示进行压缩改进前的 Device Net,压缩比为 2~8 表示改进后的 Device Net。

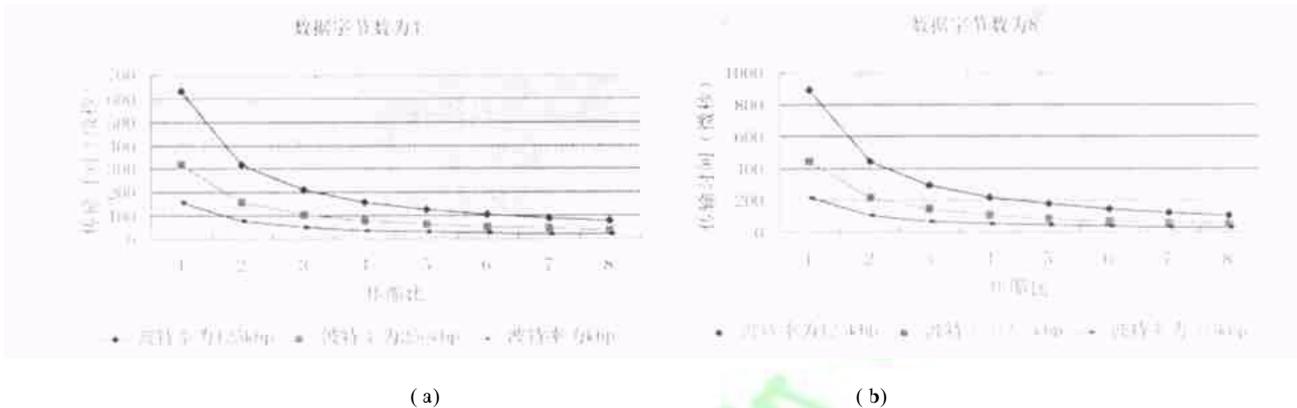


图 4 仿真曲线

Fig. 4 Simulation curves

为了便于分析,我们进行了相应的简化.设要传输的数据具有最高优先级,则其传输时间主要与系统的波特率、压缩比、有效数据字节数等因素有关.由于 Device Net 只支持三种波特率:125kbps,250kbps,500kbps,所以图中画出了三条曲线;采用新型矩形波串法对数据进行压缩,在某些情况下,压缩比过大会使重建的数据有较大的误差,因此实际应用时应根据具体情况适当选择压缩比的大小,为了不失一般性,我们设最大压缩比为 8;Device Net 中每一帧的有效数据为 0~8 个字节,这里我们只以 4 个和 8 个字节数为例进行了分析,其它情况与此类似。

从图 4 中可以看到,当数据字节数和波特率相同时,针对同一批数据,在改进后的系统中进行传输的时间明显小于改进前的系统,并且随着压缩比的增加呈现出指数下降的趋势.由此可以看出,本文所提出的改进方法可以使系统的实时性能得到改善。

5 总结 (Conclusion)

Device Net 现场总线 MAC 层采用了一种新型的“非破坏性逐位仲裁”机制,保证了数据发送的实时性和确定性.本文通过对这种控制机制的深入分析,发现非破坏性逐位仲裁正是 Device Net 现场总线速度低、扩展性差的主要原因.经过推导,找出了位时间、传输时延和采样点等参数之间的关系,并根据分析结果对 Device Net 进行了改进.首先,在它的

物理层加入跳变,实现了位同步,保证了通讯设备的安全.其次,利用数据压缩技术,减少了网络的数据流量,提高了网络的传输性能.这两种方法对 Device Net 的协议改动不大,具有良好的兼容性。

参 考 文 献 (References)

- [1] Biegacki S. The application of Device Net in process control [J]. ISA Transactions, 1996, 35(2): 9~176.
- [2] Patzke R. Fieldbus basics [J]. Computer Standards & Interface, 1998, 19(5~6): 275~293.
- [3] Open Device Net Vender Association Inc. Volume 1: Device Net Communication Model and Protocol Release 2.0, Errata 3 [Z]. 2000.
- [4] 汤同奎. Lonworks 现场总线应用研究 [D]. 上海:上海交通大学, 2000.
- [5] 李正明,等. 改进的矩形波串算法及在过程数据压缩中的应用 [J]. 江苏理工大学学报(自然科学版), 2001, 22(6): 73~77.

作者简介

方晓柯(1973-),女,博士生,讲师.研究领域为现场总线技术及其在智能控制中的应用。

顾树生(1939-),男,教授,博士生导师.研究领域为复杂系统的频域控制理论,先进控制在工业中的应用,人工智能控制理论及其应用,电力传动及其自动化等。

王建辉(1957-),女,博士,教授.研究领域为智能控制理论及应用,复杂工业过程建模与优化集成控制,基于现场总线的先进控制技术及其在工业中的应用,CAI等。