合肥工业大学
硕士学位论文
IS015765协议的研究与实现
姓名: 周涛
申请学位级别:硕士
专业: 计算机应用
指导教师: 韩江洪
2011-04

ISO 15765 协议的研究与实现

摘要

随着汽车上电控单元(ECU)数目不断增多,汽车网络化成为了汽车电子发展的趋势。CAN 总线经过长期的发展已经成为了实际的汽车底层总线标准之一。ISO 15765 作为一种基于 CAN 总线的汽车网络诊断协议,因其在传输机制、时间控制、错误检测等方面的优点,目前在汽车网络诊断中得到广泛应用。

ISO 15765 网络诊断协议对网络层的通讯时间控制较为严格,在应用层提供了较为全面的诊断服务。如何对协议中的参数进行合理的设置以减少网络层的传输时间、以及在较短的时间内针对不同的需求所选取的不同的诊断服务进行协议的实现,是目前待研究的问题。

本文在大量阅读国内外研究资料对 ISO 15765 协议分析的基础上,对 ISO 15765 协议的网络层、应用层和诊断流程进行了相关研究。通过分析 CAN 总线链路层中报文的仲裁机制以及 15765 协议的传输机制,以网络负载率、数据帧优先级参数为约束,建立了网络层中 STmin 参数模型。同传统的参数模型相比,此模型可以减少网络层在多帧传输中的平均等待时间。通过分析应用层具体诊断服务和流程,针对车载网络诊断系统的特点将有色 Petri 网和诊断系统的流程相结合,建立故障诊断的有色 Petri 网模型,描述了诊断仪与 ECU 通信之间的逻辑关系。仿真表明运用这种方法有助于提高车载网络诊断系统的软件开发效率。

最后介绍了座椅电控单元诊断系统的实现方法,并利用 CANoe 仿真软件模拟了车载网络环境,对座椅控制器的诊断系统进行了验证与分析。

关键词:CAN 总线; ISO 15765; 车载网络诊断; 有色 Petri 网

Research and realization of ISO 15765 protocol

ABSTRACT

With the growing number of electronic control unit (ECU) used on cars, automotive network turns into automotive electronic development trend. Over a long period development, Controller Area Network (CAN) has become one of the actual car bottom bus standards. As a vehicle network diagnostic protocol based on CAN, ISO 15765 is currently widely used because of its merits on transmission mechanism, time control, error detection etc...

ISO 15765 is more stringent on the time control of network layer and provides more comprehensive diagnostic services on application layer. How to reasonably set up the parameters to reduce the network layer transmission time and implement different diagnoses as per different requirements is still pending to research.

Based on lots of domestic and foreign analysis data reading and analysison ISO 15765 protocols, this thesis carries out relevant research on the network layer, the application layer and diagnosis process of the protocol. First of all, by analyzing link layer arbitration mechanism of CAN bus and transmission mechanism, Stmin parameter model of the network layer is set up which uses network load rate and data frame priority parameters as constraints. Compared with the tradition parameter model, this model could reduce the average waiting time of the network layer in a multi-frame transmission. Secondly, in accordance with the characteristic of automotivecar body network and by analyzing the specific diagnosis service and process of the application layer, the colored Petri nets model of diagnosis process is set up by combining the colored Petri nets and process of diagnosis system. This model could be used to describe the logic relationship of communication of diagnostic instrument and ECU. Simulation shows this method would improve automotive network software development efficiency.

Finally, the thesis introduces implementation of seat ECU diagnosis system, which is verified and analyzed by using CANoe simulation software to simulate the network environment.

Keywords: CAN bus; ISO 15765; Automotive Network Diagnostic; Colored Petri net

插图清单

图 1-1 诊断系统层划分	2
图 2-1 CAN 总线分层结构图	7
图 2-2 标准 CAN 报文帧结构	8
图 2-3 扩展 CAN 报文帧结构	8
图 2-4 远程帧结构	9
图 2-5 错误帧结构	9
图 2-6 过载帧结构	10
图 2-7 ISO 14230 报文结构	11
图 2-8 节点之间数据无错误的传输	14
图 2-9 节点之间数据有错误的传输	14
图 2-10 节点之间广播数据的传输	14
图 3-1 单帧传输方式	18
图 3-2 多帧传输方式	18
图 3-3 ISO 15765 的多帧传输流程	20
图 3-4 链路层数据传输延迟	23
图 3-5 标准帧的帧长度	25
图 3-6 实验仿真平台	26
图 3-7 DDCU 在 30% 网络负载率下的时间延迟(只有周期性消息)	26
图 3-8 DDCU 在 30% 网络负载率下的时间延迟(有事件性消息)	27
图 4-1 应用层时间控制	31
图 4-2 Petri 网的组成	32
图 4-3 未激发的 Petri 网	33
图 4-4 激发后的 Petri 网	33
图 4-5 Flash 下载 Petri 网建模	35
图 5-1 诊断控制单元软件分层构架	39
图 5-2 内部空间的 MEMORY	41
图 5-3 虚拟仿真系统	42
图 5-4 半实物仿真	43
图 5-5 真实物理总线仿真	43
图 5-6 CANoe 使用界面	44
图 5-7 半实物仿真图	44
图 5-8 Diagnostic Console 测试工具	45
图 5-9 Diagnostic Console 测试工具参数设置	45
图 5-10 正确的测试结果	46
图 5-11 反向错误测试	47

表格清单

表 2-1	J1939 信息帧格式	12
表 2-2	J1939 标识符分配	13
表 2-3	三种协议的比较	16
表 3-1	网络层服务原语	17
表 3-2	数据帧结构	19
表 3-3	实验中的节点	26
表 3-4	诊断网络中周期性消息实验结果	27
表 3-5	诊断网络中非周期性消息实验结果	27
表 3-6	动态改变 STmin 参数后诊断网络中周期性消息实验结果	28
表 3-7	动态改变 STmin 参数后诊断网络中非周期性消息实验结果	28
表 4-1	应用层服务分类	30
表 4-2	变迁含义	36
表 5-1	错误类型设置	46

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所

知,除了文中特别加以标志和致谢的地方外,论文中不包含其他人已经发表或撰写过的研究成果,

也不包含为获得。合肥工业大学。或其他教育机构的学位或证书而使用过的材料。与我一同工作

的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

学位论文作者签字:周涛

签字日期: 2011 年 4月 28日

学位论文版权使用授权书

本学位论文作者完全了解___合肥工业大学__有关保留、使用学位论文的规定,有权保留并向

国家有关部门或机构送交论文的复印件和磁盘,允许论文被查阅或借阅。本人授权 合肥工业大

学 可以将学位论文的全部或部分论文内容编入有关数据库进行检索,可以采用影印、缩印或扫

描等复制手段保存、汇编学位论文。

(保密的学位论文在解密后适用本授权书)

学位论文者签名:

导师签名: 韩江洪

周涛

签字日期: 2011年 4月28日

签字日期:2011年4月28日

学位论文作者毕业后去向:

工作单位:

电话:

通讯地址:

邮编:

Ш

第一章 绪论

1.1 课题的研究背景

随着人们对汽车动力性、舒适性等要求的不断提高,汽车行业的竞争也日趋激烈。伴随着激烈的竞争,汽车技术也得到了巨大的发展。自从发动机燃油系统由电子控制取代了机械控制以来,电子控制设备在汽车上的应用极大的影响了汽车的发展。越来越多的电子控制系统(ECU,Electronic Control Unit)被采用到汽车上,在某些高档轿车中ECU成本在总成本中已占到22%,2011年预计将达到40%^[1]。伴随着汽车电子技术的不断发展,尤其是汽车电子技术与计算机、网络技术的结合为汽车电子化带来了根本的变革。汽车总线作为一种计算机网络技术和工业现场总线控制技术的结合,成为了现代汽车发展的主要技术之一。汽车总线的出现,一方面使得原有的电子控制功能不断的得到了加强和细化,另一方面使得汽车上的电子控制单元越来越多,如发动机的电子控制装置、自动变速器、ABS、安全气囊、电控悬架、巡航控制系统、电控动力转向装置以及车身部分的电动门窗、电动后视镜、电动座椅等,从而使得车载网络越来越复杂。

CAN (Controller Area Network, 控制器局域网络)总线作为现场总线的重要成员是德国 Bosch 公司为解决现代车辆中众多的控制和数据交换问题所开发的通信协议。由于 CAN 总线具有数据结构较短、非破坏性总线仲裁技术以及灵活的通讯方式特点,已被众多汽车企业采用,成为汽车总线底层标准之一。根据 ISO(国际标准化组织)定义的 OSI 模型, CAN 协议定义了物理层及数据链路层规范,在实际应用中不同的汽车厂商可以根据自身需要制定更为详细的应用层协议。

各大汽车厂商根据自己的需求制定了适应自身需求的应用层协议以应对汽车行业激烈的竞争环境,然而这些协议间相互并不兼容。此外,在 ECU 的软件开发方面,各个汽车厂商同样也采取了严格的保密措施,这些做法直接导致了早期的汽车电子产品缺乏共用性和重用性。随着近几年汽车电子的飞速发展,应用需求不断增多,越来越多的资源被花费到将现有的解决方案移植到不同的车身当中。为了发挥各个厂商的优势,分工合作共同完成复杂的汽车电子系统开发,AUTOSAR、OSEK等标准被提出。

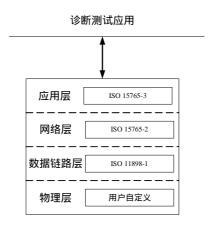


图 1-1 诊断系统层划分

AUTOSAR、OSEK 等标准只是针对汽车网络管理和整车网络构架的一种标准。随着汽车电子控制设备数量增多以及日益复杂的趋势,为满足汽车网络故障诊断系统开发的需要 基于 CAN 总线的 ISO 15765 车载网络诊断系统被提出,其系统层划分如图 1-1 所示。因此对 ISO 15765 协议的研究具有重要的意义。

1.2 故障诊断发展概况及国内外研究现状

1.2.1 汽车故障诊断协议发展概况

早期乘用车的汽车故障诊断主要是通过 K 线来实现 ,ISO 9141 和 ISO 14230 是基于 K 线的诊断协议,ISO 9141-1 是针对物理层、数据链路层、应用层三层的定义,OBD (车载诊断)则是由 ISO9141-2 定义的。诊断协议中的物理层、数据链路层、应用层的定义是由 ISO14230-1、ISO14230-2、ISO14230-3 分别进行的定义,ISO14230-4 是针对 OBD 修正的定义。ISO14230 只是定义了诊断协议大体框架,而其中包含的很多具体的内容(如故障码格式等)则必须由制造商自定义,每个制造厂商会根据不同的需求定义不同的内容。在以 ISO 14230 为大体框架的前提下,各个制造厂商则根据自身不同的需求完善、补充具体的服务内容和错误代码。所以只有获得了制造厂家所制定的具体协议内容后,才能通过诊断设备获取所希望得到的诊断信息。根据信息不同的安全需求,KWP2000 协议制定了不同的安全级别。对于普通的故障信息协议并没有设置安全保护,在知道协议的前提下可通过诊断工具读出具体的信息。而对于特定的的信息则根据信息不同的重要性,由制造厂家根据信息的不同级别进行访问限制,诊断仪必须需通过不同的安全认证才能获得这些信息[2]。

J1850 协议在 20 世纪 90 年代被美国的 GM、Ford、DC 等公司作为满足 OBD- 的诊断系统的通信标准广泛使用,但是此标准并不包含诊断协议,然后由于各种原因欧洲的汽车厂并没有使用 J1850 协议。与之相应他们设计出 ISO 15031,此协议在应用层上设计的 2、3、4、5、6 部分分别与对 SAE 的 J1930、J1962、1978、1979、2012。J1979 对应。在链路层上此协议可在 CAN-BUS、

kwp2000、ISO9141-2、J1850上运行。其中 J2190 协议是 J1979 协议的扩展,同样 J2190 协议作为诊断的扩展协议部分,只是定义了扩展部分的框架没有定义具体的服务内容。扩展的 J2186 协议对数据链路安全做出说明。目前 J1939 协议基本已经成为全球商用车所支持的应用层协议,J1939-73 也被广泛用于车身诊断中成为目前商用车主流的诊断协议。

车身诊断系统由于是独立于车载 CAN 网络的一个较为复杂的网络系统,此系统的加入使得系统成本大大增加。为了解决诊断系统和 CAN 总线融合这一问题,欧洲汽车厂商与 2000 年推出一种以 CAN 作为底层的诊断系统通信诊断标准 ISO15765,在这个诊断系统通信模型中,应用层在参考 ISO14229-1 以及 ISO15031-5 标准的内容后并兼容了大部分汽车厂商规范中所定义的诊断服务定义的诊断服务,这些服务具有测试、检查、监控、诊断管理等功能。网络层的主要作用是为应用层所制定的服务进行在网络层间的实现,其中包含了的数据帧分段、流控制帧和数据帧的重组等处理方法。物理层和数据链路层的制定则是按照 CAN 规范。总体来说在 ISO 15765 的诊断协议中,高层通过协议所制定的服务接口实现同底层之间的信息交互,低层则需要通过接口实现对上层的服务[4][5]。

1.2.2 国内外研究现状

现场总线技术起源于欧洲,经过多年的发展目前常用的总线有:基金会现场总线(FF)、CAN(Control Area NetWork)、DeviceNet、LIN等。目前在车载网络总线中应用最为广泛的是 CAN 总线,其已成为目前汽车制造业主体的行业标准,代表着汽车电子控制网络的主流发展趋势。世界上很多著名的汽车制造厂商:Volkswagen(大众)、benz(奔驰)、BMW(宝马)、Rolls-Royce(劳斯莱斯)、等都已经采用 CAN 总线来实现汽车内部控制系统的数据的传输^[14]。

今天在欧洲几乎每一辆新下线的车上均装配有基于 CAN 总线的车身控制模块。在国外,各大汽车零部件公司几乎都掌握着开发车身控制模块的技术,最有名的有美国的 Delphi,德国的 Hella,日本的 Denso 等,这些公司虽然掌握着这些核心的技术,但是技术上对外是不公开的。作为车身控制模块中的一个重要的组成部分,车身故障诊断也越来越收到了各大汽车厂商的重视,其标准也由前先的分散走向目前的统一。目前,基于 K 线诊断的 ISO 14230 和 ISO 9141成为了许多汽车厂商采用的诊断通信标准。由于 CAN 总线的渐渐成为主流的总线,基于 CAN 总线的诊断系统通信标准 ISO 15765 已被许多汽车厂商采纳,将成为目前汽车行业的通用诊断标准。在此标准基础之上汽车厂商纷纷推出了改进和扩展后的适应自己的诊断协议。

目前,国内一部分车型已采用了 CAN 总线技术,如奇瑞 A5 车型,新车型 A6 和 A3。但整体上在我国基于 CAN 整车车用网络的应用还处于起步阶段,随

着基于 CAN 协议整车车用网络的广泛应用,基于 CAN 总线的总线诊断技术已经开始引起了国内一些汽车研发部门的关注。在整车厂的推动下,基于 CAN 总线的故障诊断技术将加快普及和发展,该技术将成为今后汽车诊断技术的主流。因此本课题研究的内容符合当前车用总线诊断技术发展的趋势,具有重要的现实意义。

1.3 本课题的来源

课题来源于广东省教育部产学研结合项目"基于 CAN 的车用网络系统集成协议及系统开发"(编号:2008B090500226)。高等学校博士学科点专项科研基金"安全关键分布式实时系统设计方法及关键问题研究"(编号:20070359028)。

1.4 研究内容及组织结构

1.4.1 本文研究内容

ISO 15765 总共分为 4 个部分: ISO 15765-1 物理层及数据链路层说明、ISO 15765-2 网络层的说明、ISO 15765-3 应用层的说明、ISO 15765-4 排放相关说明。本文主要针对协议的网络层和应用层进行研究分析。对其协议网络层的传输机制进行了研究和改进,对应用层服务流程和应用层的具体服务进行了建模。在此基础上实现了座椅诊断控制单元,利用汽车网络仿真工具 CANoe 进行了半实物仿真验证。

1.4.2 本文的组织结构

第一章介绍了本课题研究的背景和国内外的研究现状。在汽车网络应用的 背景下引出对汽车诊断系统研究的必要性。

第二章对汽车电控单元诊断系统的知识进行分析,包括 CAN 总线和目前几种常用诊断协议,并对这几种诊断协议进行比较和分析。

第三章对诊断系统中网络层所选用的 ISO 15765 诊断协议进行分析。从而找到协议中降低传输延迟和提高传输稳定性重要的 STmin 参数。通过对 STmin 参数进行具体的分析,计算其数学期望后得到了一种动态设定算法。通过实验证明此方法是可以有效的改善诊断协议中网络传递的

第四章首先对诊断系统中应用层的总体框架进行分析,针对车身诊断系统的特点,通过对诊断系统应用层的分析,选取了有色 Petri 网作为建模工具,描述了诊断服务之间的逻辑关系,提出了一种建模的方法。最后仿真证明该模型的控制逻辑正确,应用该方法有助与提高车身控制系统的软件开发效率。

第五章重点讨论了车身诊断系统的的构建方案。首先设计了诊断系统的分层结构,给出了软件设计框架,通过 CANoe 对所设计的汽车座椅诊断系统进行

测试,证明其可行性。

第六章对全文的工作进行了总结,分析了文章的创新点和尚未完善的地方,提出了进一步的工作。

第二章 CAN 协议及诊断协议的分析

首先对汽车网络总线 CAN 进行了介绍与分析,对其分层结构和报文类型及错误处理进行了重点介绍。同时对目前比较成熟的可用于车身诊断的成熟的通信协议 ISO 14230、SAE J1939、ISO 15765 进行了的分析,并比较了他们的优缺点。

2.1 CAN 总线技术介绍

CAN 总线是德国 BOSCH 公司 1985 年为解决汽车中诸多测量装置与控制装置之间的信息交流而开发的一种支持分布式控制与实时控制的串行通信网络。随着 CAN 技术应用的不断推广, BOSCH 公司于 1991 年提出了 CAN 技术规范 2.0 以便于进行通信协议的标准化工作,该规范包含 A、B 两部分;其中2.0A 给出了标准报文格式(11 位标识符),2.0B 给出了标准报文格式和扩展报文格式(29 位标识符)。国际标准化组织 ISO 于 1993 年提出了 CAN 国际标准ISO11898,为 CAN 进一步的标准化和规范化铺平了道路,促进了 CAN 的快速发展^[28-33]。其特点为:

- 1. 通信方式灵活,其工作方式为多主节点。只要网络空闲,网络上的任意 节点都可以主动的向网络上的其他节点发送信息。
- 2. CAN 网络上的节点信息,根据消息的不同设置不同的优先级,用以满足不同的实时需求。
- 3. 数据帧短,反应速度快。
- 4. 使用价廉物理介质。
- 5. 多个节点可以同时发送,得到仲裁数据获取总线,并且具有远程数据请求功能。
- 6. 能判别暂时错误和永久性错误节点,具有故障节点自动脱离功能。
- 7. 设置 CRC 校验位以及及其它检错措施以保证数据通信的可靠性。
- 8. CAN 总线通信接口中集成了 CAN 协议的物理层和数据链路层功能。

2.1.1 CAN 总线的分层结构

CAN 定义了 ISO/OSI 网络系统模型的物理层和数据链路层。其中主要是数据链路层进行了定义。在不同的版本中,分层的定义有一些区别。CAN2.0 的 ISO/OSI 模型分层结构如图 2-1 所示。

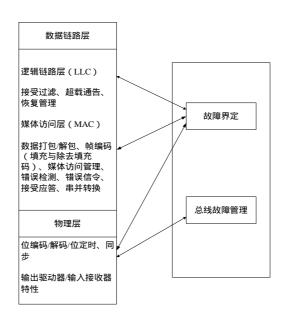


图 2-1 CAN 总线分层结构图

- 1.物理层:物理层(Physical Layer)是将 ECU 连接至总线的电路实现,信号是如何实际地传输。BOSCH CAN 几乎没有对物理层定义。同样基于 CAN 网络,不同的物理层可能差别很大,但同一网络上的所有节点物理层必须相同。
 - 2.链路层:数据链路层又分为逻辑链路控制(LLC)和媒体访问控制(MAC)。
 - (1) 逻辑链路控制(LLC-Logic Link Control)的主要作用是:
 - 接收滤波、超载通告和恢复管理。
 - 作为同 LLC 实体(LPDU)之间进行交换的数据单元,包括 LLC 数据帧和远程帧。
- (2) 介质访问控制子层 MAC(Medium Access Control)是 CAN 协议的核心。它把接收到的报文提供给 LLC 子层,并接收来自 LLC 子层的报文。MAC 子层的主要作用是:
 - 负责报文分帧、仲裁、应答、错误检测和标定。
 - 根据自检测机制,区别永久故障和短时扰动。

2.1.2 CAN 总线的传送和帧结构

CAN 总线报文在通信过程中是以帧为单位进行传输,为了实现不同的功能,规定了数据帧、远程帧、错误帧、过载帧这 4 种不同的帧类型。

1. 数据帧

用于将数据从发送器传输到接收器。数据帧由帧起始、仲裁场、控制场、数据场、CRC 场、应答场和帧结束这 7 个不同的位场组成,其中数据场长度可以为 0。数据帧有标准帧和扩展帧两种如图 2-2、2-3 所示。

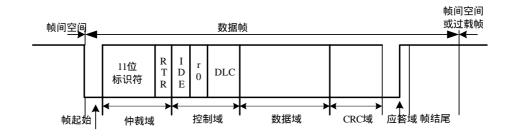


图 2-2 标准 CAN 报文帧结构

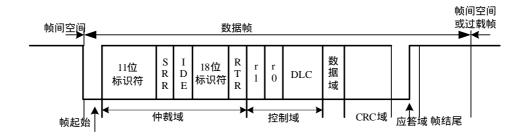


图 2-1 扩展 CAN 报文帧结构

在 CAN 2.0B 中有两种不同的帧格式,根据表示符的长度不同可以分为标准帧(11 位标识符),扩展帧(29 位表示符)

(1) 帧起始

帧起始标志数据帧和远程帧的起始,由一个单独的"显性"位组成。只有当总线空闲时,才允许节点开始发送。所有的节点必须同步于首先开始发送信息节点的帧起始前沿。

(2) 仲裁场

用于报文在发送和接收过程中的总线仲裁和滤波;标准帧和扩展帧的仲裁域格式不相同。标准帧的仲裁域由 11 位标识符(ID28—ID18)以及 RTR(Remote Transmission Request Bit, 远程发送请求位)位组成;其中 RTR 位、IDE 位在此需为显性位(见图 2-2)。

扩展帧的仲裁域由 29 位标识符(ID28—ID0)、SRR(Substitute Remote Request Bit,替代远程请求位)位、IDE(Identifier Extension Bit,标识符扩展位)位及RTR位组成;其中RTR位需为显性位,SRR位、IDE位需为隐形位(见图 2-2)。

(3) 控制场

用于控制数据帧里面的数据长度(0 到 8 字节)。由两个保留位 r0 ,r1 或 IDE 和四位数据长度码组成。

(4) 数据场

用于填充数据帧里面待发送的数据,长度为0字节到8字节。

(5) CRC 场

用于对数据帧进行循环检测,由 CRC 序列和 CRC 界定符组成。

(6)应答场

用于在接收到数据帧时向发送器发送一显性位用于应答,由应答间隙和应答界定符组成。

(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。

2. 远程帧

总线上任意节点在任意时刻均可通过向总线上其它节点发送远程帧来请求其发送具有相同标识符的数据帧。由于该帧不是用来传输数据的,故其由 6部分组成,仅比数据帧少了数据域,其中 RTR 位为隐形位(见图 2-4)。

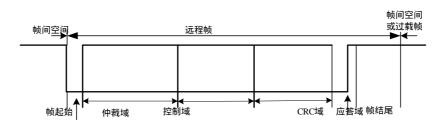


图 2-2 远程帧结构

3. 错误帧

错误帧用于检测到总线错误的总线单元发送错误信息,由错误标志及错误 界定符组成。错误标志是不同总线节点提供的错误叠加,分为"激活错误"标志(6 个连续显性位)和"认可错误"标志(6 个连续隐形位)。每个"错误认可"节点为了正确终止错误帧,要求总线必须有至少三个位时间的空闲,故总线负载需小于 100%(见图 2-5)。

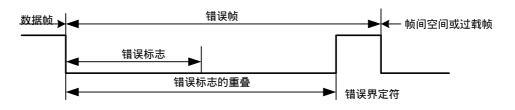


图 2-3 错误帧结构

4. 过载帧

用于为相邻数据帧或远程帧提供附加的延时,由过载标志(6个显性位)和过载界定符组成。总线节点检测到以下三种情况时会发送过载标志:一是由于接收器的内部原因需要延迟下一个数据帧与远程帧;二是在间歇(帧间空间的)的第一位与第二位检测到显性位;三是总线节点在过载界定符或错误界定符的最后一位检测到显性位(见图 2-6)。



图 2-4 过载帧结构

2.1.3 错误类型和故障处理

CAN 总线在运行过程当中会产生五种类型的错误。

1. 填充错误

总线节点发送器在任意时刻检测到帧起始、仲裁域、控制域、数据域和 CRC 序列等部分有五个连续相同的位值(00000 或者 11111),将会自动插入一个相反的位值(000001 或者 111110)。若总线节点检测到 6 个连续一样的位值(000000 或者 111111),便会产生填充错误。

2.格式错误

在标准格式或扩展格式下各类型帧的位数都是确定的,若某个部分多出一位或几位,便会产生格式错误。

3.位错误

总线节点在发送信息帧时,会对总线进行监视。若监测到的位值与正在发送的位值不同,便会产生位错误。

4. CRC 错误

总线报文在发送过程中,会通过循环冗余码对信息帧进行检测。若发现接收器按照循环冗余码计算的结果与发送器计算的值不同时,便会产生 CRC 错误。

5.应答错误

总线节点成功接收有效信息帧后,会在应答间隙期间向发送器发送一显性 位以作应答。若总线节点监测到的不是显性位时,便会产生应答错误。

总线任意节点检测到填充错误、格式错误、位错误和应答错误中的任一个后,便会在下一位发送"错误标志"; 当检测到的为 CRC 错误时,便会在应答界定符之后的位发送"错误标志"。

2.2 常用的汽车诊断协议

CAN 规范 2.0 只定义了 OSI 网络模型中的物理层(第一层)与数据链路层(第二层),这两层通过硬件(CAN 控制器与发送器)来完成;没有对与实际应用密切相关的应用层(第七层)做任何规定,不是一个完整的协议。需要用户在实际应用过程中引用成熟的通信协议或根据系统需求制定相应的通信协议^[22]。

现有的国际上比较成熟、实用的应用层可用于诊断的协议有 ISO 14230、SAE J1939 和 ISO 15765 等,它们都是针对特定应用领域所开发的,均有独特的设计思路与组成特点。

2.2.1 ISO 14230 协议

1.物理层

ISO14230 - 1 协议作为诊断协议部分的物理层,是基于 ISO 9141 - 2、ISO 9141 , 道路车辆诊断系统数字信息交换技术要求,他的波特率定为 10.4~kb/s , 使用的是单线 K 线通信,但也可用双线 K 线和 L 线通信。根据不同的需要有的 ECU 不需要初始化就可直接通信,K 线本质上是一种半双工串行通信总线。

2.数据链路层

ISO14230 的数据链路层需求在 ISO14230 - 2 中定义 格式如图 2-7 所示。

	报文头				数据域			校验和
格式	目的地址	源地址	长度	数据1	数据2		数据N	校验和

图 2-7 ISO 14230 报文结构

在 ISO 14230 中数据链路层所需要完成的功能将从应用层传来的数据打包 传给物理层,主要包括两部分工作:

- (1)在报文中加报文头;
- (2)在报文尾加校验和。

按照协议规定报文头中首字节的高两位表示的内容是类型,而后面的低六位表示的是报文数据域的长度,其中节为 0 时则用报文头的第 4 字节表示数据域长度,其中能够传输的报文长度最长为 255 字节。第 2 个字节表示目标地址,源地址由第 3 个字节表示,数据域部分存放的是应用层传来的数据,校验和位按照累加和算法进行计算。

3. 网络层

ISO 14230 中没有对网络层进行定义。

4.应用层

在 ISO 14230 诊断协议中应用层就是有关诊断的应用,通过事先同诊断仪约定好诊断号。以及诊断码所代表的诊断内容,其数据格式如下:

服务号[+子功能][+参数列表][+数据]

上式的方括号表示可选。

2.2.2 SAE J1939 协议

SAE J1939 协议是美国汽车工程师协会为了实现总线协议的标准化,在

CAN2.0B 的基础上制定的应用于大型货车和客车的车辆网络串行通信和控制协议^[19]。该协议是按照 ISO/OSI7 层参考模型建立的,在物理层和数据链路层基本上沿用了 CAN 规范,并增加了网络层、应用层和网络管理规范,网络通信速率可达 250Kbps。J1939 协议因其独特的设计理念和优异的性能广泛应用于货车、客车、卡车、农业装备、林业电子设备、柴油机控制、海洋电子设备、公共交通、铁路等等。

1.物理层

J1939 协议的物理层是实现总线上电控单元的连接(单网段最多为 30 个,取决于总线驱动能力),规定信号如何进行传输的;主要解决 4 方面问题:数据传输的物理媒介参数、位电平、位同步的实现、总线各节点的硬件构造。

- (1)本协议规定的物理介质有屏蔽双绞线和非屏蔽双绞线,其特定阻抗为 120 ,两条线分别为 CAN_H 和 CAN_L ,分别对应电压 VCAN_H 和 VCAN_L , 彼此间存在差动电压 V_{diff}。
- (2)总线始终位于显性(逻辑 0)或者隐形(逻辑 1)状态之一。在总线发生冲突时,总线节点发送器在发送信息位之前会先将位电平与总线检测电平进行比较,若相同(同为显性或隐形),则继续发送该位,反之若发送的为"隐性",而监测到的为"显性",该发送器将不能继续发送;可在总线空闲时发送或下次仲裁时重新进行比较。

2.数据链路层

J1939 协议将 CAN2.0B 的扩展帧(29 位)格式进行重新定义,对其 29 位标识符进行合理分配(见表 2-1、2-2)。其中 RTR 位需为显性位, SRR 位、IDE 位需为隐形位(取为 1)。标识符是总线非破坏性仲裁和报文接收滤波的主要依据,其主要包括优先级 P(Propriety)、保留位 R(Reserve)、数据页 DP(Data Page)、协议数据单元格式 PF(Protocol Formula)、协议数据单元特性位 PS(Protocol Special)和源地址 SA(Source Address)。

帧起 始		仲	裁域				控	制域	数据域	校验域	应答 域	帧结束
SOF	11 位标 识符	SR R	I DE	18 位 标识 符	R T R	r o	r 1	DLC(数 据长度 代码)	0-8 字 节数据	16 位 CRC	2 位 应答 域	E 0 F

表 2-1 J1939 信息帧格式

表 2-2 J1939 标识符分配

标识符功能	优先权 (P)	保留位(R)	数据页位 (DP)	协议数据单元 格式(PF)	协议数据单 元特性位 (PS)	源地址 (SA)
7+ 応 1D	ID. 1-	ID.4(取	ID.5(取	ID. 6- ID. 13	ID. 14- ID. 21	ID. 22-
对应 ID	ID. 3	0)	0或1)	ט.ט- וט.וט	ID. 14- ID. 21	ID. 29

3. 网络层

数据链路层的另一重要功能就是传输协议功能,它主要包括两方面:消息 的拆装与重组以及连接管理。

- (1)当传输的数据帧中的数据大于 8 字节,就必需采用多帧的形式进行信息传递,这就需要将多帧进行拆分成长度小于 8 字节的数据包;为了在拆分以后较好的区别各个数据帧的位置,将数据场的第一个字节用于对各拆分的数据帧进行数据编号,将拆装的各数据帧按照拆分顺序进行依次排号;目的节点在将这些信息帧完全接收以后,按照编号顺序将各个帧数据组合为原始数据。第一字节是 8 位二进制,从 1 到 255 都可以用于数据编号,故每多帧数据最多能传送 7x255=1785 字节的数据。另外,在传输各个数据帧时,各帧信息之间的间隔对于特定目的地址的信息不能长于 200 毫秒,对于全局信息间隔最好为 50—200毫秒(见图 2-5、2-6、2-7)。
- (2)连接管理用于节点之间进行数据传输时虚拟连接的建立与关闭,分为两种情况:广播信息与发送到特定地址的信息;对于特定信息来说,发送节点需要先向目的节点发送一个请求信息(TP.CM_RTS),用于告诉该节点有节点准备与之建立通信连接,接收节点如果准备好需向发送节点发送一个准备发送信息(TP.CM_CTS),用于告知发送节点,其已经做好信息接收准备;这时节点之间的连接就建立起来了,接下来发送节点就可以发送各个数据帧,在最后一帧信息发送完毕后,发送节点需发送一个消息结束应答信息(TP.EndofMsgACK),超过一定时间后连接会断开;若发送双方在传输过程中任一方出现故障(超时、故障、资源不足等)均可发送放弃连接信息(TP.Conn_Abort)来断开连接(见图 2-8、2-9)。
- (3)对于广播信息,发送节点只需向广播节点发送一个请求信息,用于告诉该节点有节点准备与之建立通信连接,不用进行其它操作即可传输数据(见图 2-10)。

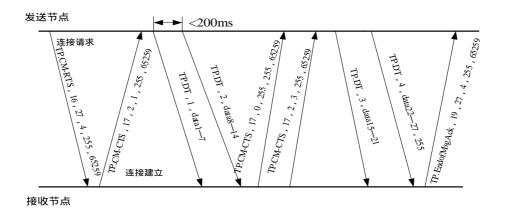


图 2-8 节点之间数据无错误的传输

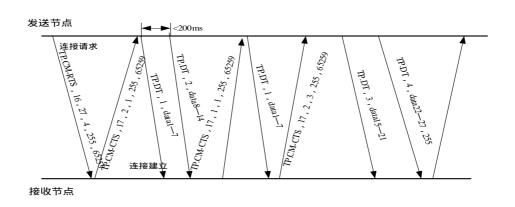


图 2-9 节点之间数据有错误的传输

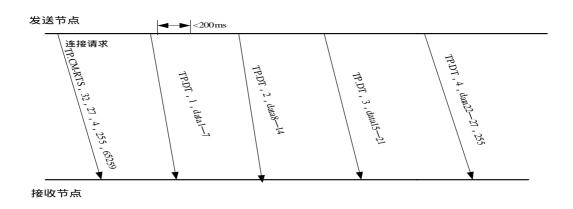


图 2-10 节点之间广播数据的传输

4.应用层

该协议在应用层对消息格式中的参数及参数组进行了详细的定义和规范的

说明,对参数数值范围和新参数数值范围进行了规定;新参数的增加需从源地址、目的地址、信号类型、数据长度、刷新率、数据分辨率及其取值范围几方面进行规定,并定义了诊断信息及标定信息。

2.2.3 ISO 15765 协议

ISO 15765 协议是一种基于 CAN 总线上的诊断协议。该协议也是按照 ISO/OSI 7 层参考模型建立的,其中 ISO 15765-1 包括了物理层和数据链路层, ISO 15765-2 对网络层进行了说明,ISO 15765-3 则是规定了应用层的具体服务。目前 ISO 15765 协议已经广泛的使用在轿车、汽车控制工业等方面。

1.物理层

ISO 11898-1 标准是 ISO 15765 协议的物理层和数据链路层所遵循底层的标准。通过使用 CAN_H 和 CAN_L 的差分电压来表示 CAN 总线中的显性和隐性位,可以用数据的 1 和 0 来表示。

2.数据链路层

根据 ISO 15765-1 中的定义。链路层的功能是把通过网络层传来的诊断报 文消息,经过修改打包成能够在 CAN 总线上进行传输的 CAN 帧。

ISO 15765 的数据层在 CAN 总线上传输时可以使用标准帧(11 位标识符), 遵守 CAN2.0 A, 11 位标识符定义了 CAN 节点在总线上的唯一的地址。报文的优先级是按照标识符从高位到低位按位进行仲裁 0 的优先级高于 1。同样也可以使用扩展帧(29 位标识符)遵守 CAN2.0B, 其数据帧的格式就是利用 CAN的数据帧进行传输。

3. 网络层

根据 ISO 15765-2 中的定义,网络层的功能是接收到应用层发送过来的消息流后,根据定义中的分包、位填充和时间控制等步骤,对消息流进行控制传输。流控制输有单帧传输、多帧传输两种类型,可根据传输数据的长度进行帧传输方式的选择。

相反当接收来自诊断仪数据时,网络层则完成的是一个是组包的过程,这时需要一个可以存放接收的数据的缓冲区。单帧的组包相对简单,而接收到多帧时数据则需要根据接收到的首帧和连续帧的数据域中提取,后按照排列顺序重新组合交给上层的应用层。

传输具体的 ISO 15765 协议网络层分析将在第三章第一节进行了分析。

4. 应用层

在汽车诊断协议中应用层就是有关诊断的应用 ,ISO 15765 协议的应用层内容在第四章第二节进行了详细的分析。

2.3 三种诊断协议的比较

通过 2.2 小节对目前使用诊断协议的设计思路及组成结构的详细分析,可

发现该类协议各具特点,分别适用于不同的应用领域。本节就基于的协议、通信方式、传输模式、功能划分、应用领域等方面对比分析几种协议。

表 2-3 三种协议的比较

	ISO 14230	SAE J1939	ISO 15765
标识符分 配方式	只支持标准帧	支持标准帧和扩展帧格式	支持标准帧和扩展帧格式
通信模式	物理	特定目的/广播/私有通信	物理/广播
数据域最 大长度	8 字节	8 字节	无限制
使用的底 层总线	K 线	CAN 总线	CAN 总线
过程数据 交换对象	协议数据单元 PDU	协议数据单元 PDU	协议数据单元 PDU
连接建立	发送连接请求	发送连接请求	可以直接发送
节点数目	最多 127 个	最多 30 个	最多 127 个
网络管理 机制	规定 ECU 的类型和地址	规定 ECU 的类型/地址声明/分配	规定 ECU 的类型和地址
主要应用领域	早期的轿车、乘用车系统	货车、客车、卡车、农业 装备、林业电子设备、柴 油机控制、海洋电子设备、 公共交通、铁路	轿车、汽车工业控制系统

ISO 14230 协议主要是基于 K 总线的车身诊断协议 ,SAE J1939 协议主要应用于货车、客车、卡车。目前基于 CAN 总线上最常用的汽车诊断协议为 ISO 15765 协议 ,它更大限度的发挥了 CAN 的优异性能并具有很高的可靠性。因此在实现诊断协议前需要对 ISO 15765 协议进行详细的研究。

第三章 ISO 15765 网络层中参数 STmin 的动态计算方法

针对诊断协议网络层中如何提高数据帧传输的稳定性并减少传输时间的问题,通过对 ISO 15765 协议网络层的分析得出 STmin 参数的取值直接关系到多帧传输时的传输延迟。以网络负载率、数据帧优先级参数为约束,在对通信中的队列延迟和传输延迟综合分析的基础上建立了 STmin 参数模型,并给出了最小平均时延的计算公式。通过仿真实验验证,与传统的定值方法相比该模型所获得的动态 STmin 参数值能更好的降低数据帧的传输时延。

3.1 ISO 15765 网络层协议分析

ISO 15765 网络层的功能主要有:

- (1) 为上层的应用层提供接口服务。
- (2)进行内部操作。为上层服务提供接口从而实现数据交流;主要有为对 等实体间的通信提供分段、重组和流控制这几种方法方法。
 - (3) 网络层中的时间控制。

3.1.1 网络层原语服务分析

对于为上层提供服务,网络层一共定义了以下 3 个原语服务,如表 3-1 所示。

服务原语	作用
要求原语服务	由发送端上层使用。用于向网络层传递控制信息和
安水原后服另	诊断服务数据
确认原语服务	由发送端网络层使用。用于向上层确认所要求服务
州 () () () () () () () () () () () () ()	的执行结果(成功或失败)
指示原语服务	由接收端网络层使用。用于向上层指示数据接收的
1	执行结果(成功或失败)

表 3-1 网络层服务原语

通过这三种原语服务可以与上层服务的建立起简单通信,对于这些网络层的原语服务都遵循如下结构:

Server_name.type
{
 N_AI
 N_PCI
 N_Data
}

其中 Server_name: 是指具体的服务(在应用层中的规定); Type: 三种原

语服务中的一种。

N_AI 包含了原语服务的地址信息,一般包含:是否是远程帧(Mtype)原语的发出地址(N SA)、原语的目的地址(N TA)。

N_PCI 则包含了原语帧的具体信息如:此消息帧属于何种帧类型(单帧、 首帧、连续帧)帧的长度(Length)等,在后面会详细介绍。

N Data 为该原语所包含的具体的数据。

3.1.2 网络层内部传输服务

在内部传输操作中,由于 CAN 总线上的数据帧每帧只能传输 8 个字节, ISO 15765 为了适应 CAN 总线上的数据传输,根据诊断报文数据长度的不同规定了单帧传输和多帧传输这两种方式。

1.单帧传输

当应用层的诊断服务数据可以利用单帧 CAN 数据传送时,可以采用单帧 传送方式,如图 3-1 所示。

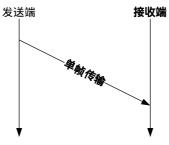


图 3-1 单帧传输方式

2. 多帧传输

当所需传输的数据无法利用单帧 CAN 报文传送时,则使用多帧传送方式进行。在多帧传送方式中,网络层根据需要,将诊断数据进行拆分成一个首帧和多个连续帧。首帧包含了分段数据的总长度信息以及一些数据帧;每个连续帧的第一个字节都包含拆分的顺序编号,后面的七个字节用于存放诊断数据。接收端在接收到连续帧后根据接收数据帧的编号重组服务数据。多帧传送方式如图 3-2 所示。

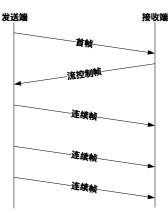


图 3-2 多帧传输方式

发送端和接收端通过 N_PCI 确认发送或者接收的帧为首帧、流控制帧还是连续帧, N_PCI 的具体结构如表 3-2 所示:

N_PCI byte N PDU 名 第1个字节 第 4-8 个字 字 第2个字节 第3个字节 高4位 低 4 位 节 有效数据字 单帧 N PCItype=0 DATA1 DATA2 节数 多帧中所包含的有效数据字 多帧首帧 N_PCItype=1 DATA1 节数 多帧连续帧 N_PCItype=2 SN DATA1 DATA1 流控制帧 BS N PCItype=3 FS STmin

表 3-2 数据帧结构

接收端通过对接收到数据帧的第一个字节进行分析就可以得出所接收到的数据帧为何种数据帧,以及所需要接收的数据长度。

接收的数据第一个字节的高四位为 0 则表示所接收的数据帧为单帧,再通过读取低四位以确定单帧数据中的有效数据字节长度。

接收的数据第一个字节的高四位为 1 则表示所接收的数据帧为多帧的首帧。再读取第四位和第二个字节后确认所要接收的字节数后,根据接收的能力和双方通信速率发送流控制帧,然后等待多帧连续帧的到来。

接收的数据第一个字节的高四位为 2 则表示所接收的数据帧为多帧的连续帧。然后在比对 SN(连续帧编号)之后将数据放入接收缓冲区。

接收的数据第一个字节的高四位为 3 则表示所接收的数据帧为流控制帧。 低四位的 FS 表示发送端所需要进行的动作。BS 为能够连续发送连续帧的数量、 STmin 为每两帧之间的时间间隔。

3.1.3 网络层时间控制分析

网络层的时间管理是为了保证发送端和接收端不会因为等待而永久的挂起,从而失去了通信的能力。因此在每次通话的过程中设置了超时的时间参数,如图 3-3 所示。

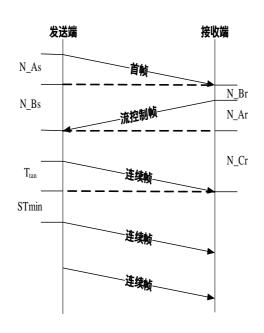


图 3-3 ISO 15765 的多帧传输流程

在网络层传输的过程中一般涉及到如下几个时间超时参数见图 3-3,他们的含义是:

N_As:发送端将数据传送到接收端的最大时间。

N_Bs:发送端在成功发送首帧后到收到流控制帧的最大时间。

N Br:接收端在接收到首帧后到发送流控制的最大时间。

N Ar:接收端将流控制传送到发送端的最大时间。

N_Cr:接收端在成功发送流控制后到收到连续帧的最大时间。

T_{tan}:每帧传输的时间。

STmin:连续帧两帧之间的最小发送间隔。

根据整个的传输流程我们可以得到诊断报文的传输延迟的公式:

$$T = N_A s + N_B s + N_C r - T_{tan} + (T_{tan} + ST min)N - ST min$$
 (3-1)

 N_A s、 N_B s、 N_C r、 T_{tan} 在实际应用中均为固定值,带入式 3-1 可知在多帧传输中,STmin 参数对于多帧传输的时间有着重要的影响。

3.2 排队论模型

基于 CAN 网络的诊断协议网络层中,我们可以将整个网络看成一个队列,我们可以将网络上传输的消息以到达率为 泊松分布。如此我们就可以使用排队论来分析数据网络的延迟,本节将对两种排队论模型进行介绍。

3.2.1排队论模型-里托定理

在排队论中,我们认为客户在随机的时间到达并获得服务。在数据网络中,顾客代表了分配给每个节点需要的传输的数据包。顾客得到的服务时间就是数据包的传输时间。为了方便起见,我们在本节中认为数据包的长度包括了帧头

和帧尾的信息。

特别值得注意的是下面 3 个量:

- (1) 网络中的平均数据帧的数目
- (2)每个数据帧的平均延迟时间
- (3)数据帧的到达率

假设我们对一个样本系统从时刻 t=0 到 t= ,并进行观测随时间的增长记录下感兴趣的特别是定义:

N(t)= 时刻 t 时网络中数据帧的数量;

A(t)=在时间段[0,t]中到达的数据帧的数量;

Ti=第 i 个数据帧在网络中花费的时间。

我们将到 t 时刻为止,网络中数据帧数目的典型值直观地表示为:

$$N_i = \frac{1}{t} \int_0^t N(\tau) d\tau \tag{3-2}$$

我们将 N_t 称为 N()从时刻 0 积累到时刻 的时间平均值,很明显, N_t 的值将随着时间 t 的改变而改变,但是在许多我们感兴趣的系统中,我们发现随着时间的增加, N_t 趋向与稳态值。

$$N = \lim_{t \to \infty} N_t \tag{3-3}$$

在这种情况下,我们将 N 称为 Nt 的稳态时间平均值(或称为时间平均值)。同时我们将称为到达时间端[0,t]上的平均值。如果极限存在累计到时刻 t 的数据消息延迟平均延迟时间可类似地定义为:

$$T_{t} = \frac{\sum_{i=0}^{a(t)} T_{i}}{a(t)}$$
 (3-4)

根据以上的知识,我们可以用一个简单的公式将 N、 和 T 三个量彼此联系起来,这样,我们在已知一个量的条件下就可以推出另外一个量。这个公式就是里托定理:

$$N=T$$
 (3-5)

3.2.2 M/G/1 队列系统

在单通道的网络队列系统中,当数据帧到达率为 泊松分布,数据帧的服务时间是普通的概率分布(不需要呈指数分布)。假设数据帧先来先服务,并且 X_i 表示第 i 个到达的数据帧的服务时间,随机变量(X_1 , X_2 , ...) 具有相同的概率分布且相互独立,而且到达时间间隔也是相互独立的,因此有式(3-6)成立。

$$W_i = R_i + \sum_{j=i-N_i}^{i-1} X_j$$
 (3-6)

式中 W: 为第 i 个数据帧在队列中的等待时间。

 R_i 为当第i 个数据帧到达的时候,数据帧j 还在接收服务,则 R 就是数据帧j 完成剩余服务所需要的时间。如果第i 个数据帧到达的时候没有其他数据帧正在处理,则 $R_{i=0}$ 。

X; 为第 i 个数据帧的服务时间。

N_i等于当第 i 数据帧到达的时候, 队列中数据帧的数目。

对公式(3-6)两边求预期值,再利用随机变量 N_i 和 $X_{i\text{--}1}$, ... , $X_{i\text{--}n}$ 的独立性 我们得到:

$$E\{W_i\} = E\{R_i\} + E\{\sum_{j=i-N_i}^{i-1} E\{X_j \mid N_i\}\} = E\{R_i\} + \overline{X}E\{N_i\}$$

当 i 趋于 时可得:

$$W = R + \frac{1}{\mu} N_{\mathcal{Q}} \tag{3-7}$$

这里 R 为平均残留的时间。在式(3-7)中,所有长时间的平均值都看成当时间和顾客数量趋向无穷大时的极限值。因此 W、R 和 N_Q 分别表示第 i 个数据帧的平均等待时间、残留时间和队列中的数据数目。这些极限值都存在,且基本上所有的重要系统中都有 $< \mu$ 时。根据里托定理有:

$$N_0 = W$$
 (3-8)

将此公式带入公式可知:

$$W = R + \beta W \tag{3-9}$$

这里 W 为 / µ 表示利用因子得到:

$$W = \frac{R}{1 - \beta} \tag{3-10}$$

最后将关于 R 的表达式代入式(3-10)就可以得到平均所需要等待的时间:

$$W = \frac{\lambda \overline{X^2}}{2(1-\beta)} \tag{3-11}$$

上述推导基于以下两个假设:

- (1)存在稳态的平均值 $W \setminus R$ 和 N_0 。
- (2) 最后的等式长期时间平均值等于响应的整体平均值。

3.3 STmin 参数模型的建立

根据本章第一节对 ISO 15765 网络层的协议分析,我们可以知道 ISO 15765 在网络层传输的过程中有两种方式:单帧传输和多帧传输。单帧传输的过程中 发送数据帧只需获取总线的仲裁后即可发送数据,而多帧传输的过程中,发送 端需要不停的发送连续帧。如何通过对 STmin 参数的设定来适应不同负载率下 的总线网络是本节所需要解决的问题。

3.3.1 网络层中数据的延迟模型

排队论是分析网络延迟的主要理论框架,运用排队论的时候,贴近实际的假设会使分析变得非常困难,所以通常我们需要做一些简化的假设。在下面本节中的我们主要集中的讨论通过网络层中的数据包延迟。按照次序,链路层的延迟包含了4个部分如图 3-4 所示:

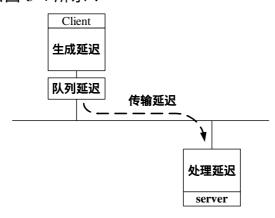


图 3-4 链路层数据传输延迟

- (1)处理延迟:指当数据包正确到达链路的头节点与该数据包被分配到一个发送队列之间的时间差。
- (2)队列延迟:指当数据包进入发送队列与该数据包开始被传输之间的时间差。在这一时间段中,该数据包将等待比其先到达队列的数据。
- (3)传输延迟:指当开始传输数据包的第一个比特位到传输完最后一个比特位之间的时间差;
- (4)生成延迟:从发送节点处理器接收到本节点的请求,到它将准备好的数据写入缓存队列里的时刻。

在我们估计的传输延迟过程中我们忽略了数据包因为传输错误或者其他原因而引起的重新传输的可能。由于随着处理器的强大按照现在的微处理器水平,生成延迟和处理延迟可以忽略不计。因此在数据网络中的延迟可以用队列延迟 (T_m) 加传输延迟 (C_m) 来表示:

$$T=T_{m}+C_{m} \tag{3-12}$$

3.3.2 STmin 参数期望模型建立

根据第一节得到的公式(3-1)可知,T 为诊断报文的期望传输延迟,T_{tan}为将诊断信息帧成功发送到缓冲区的时间,按照现在的微处理器水平此时间间隔可以忽略不计。STmin 为成功的将诊断信息发送到缓冲区后到将下一帧诊断信息发送到发送缓冲区的时间间隔。N 为所需发送的帧数,其值可由下式计算。

$$N = \frac{L_{data} - 6}{7} + 2 \qquad L_{data} > 8 \tag{3-13}$$

由于多帧传输中每帧的字节长度不能改变,即每个数据包所包含的帧数 N 也是固定的。可见 STmin 在诊断报文的传输延迟 T 中是重要的一个因素。如果 STmin 设置过小,理论上可以减小整个的传输延迟。但是如果此刻网络上的负载率已经很高,突然大量报文的发出很有可能会引起网络堵塞以至报文出错甚至丢失。如果 STmin 设置过大,则会大大影响了诊断报文的实时性。为了更好的对 STmin 进行设置,我们做了如下的分析。

STmin 是指:将一帧诊断信息从微处理器的发送缓冲区发送到对等实体的接收缓冲区的时间。故 STmin 可以用队列延迟(t_m)加传输延迟(C_m)来表示:

$$STmin=t_m+C_m \tag{3-14}$$

1. 网络层中队列延迟(t_m)的计算

队列延迟可以分为仲裁延迟和非仲裁延迟两部分。仲裁延迟指两帧消息同时发送时,高优先级消息帧造成低优先级消息帧的延迟,它包括在节点内部的仲裁延迟和总线上的仲裁延迟两部分,当队列缓存较大时,可以将节点内部的仲裁延迟和总线上的仲裁延迟合为一体考虑。非仲裁延迟是由于总线上已有其他消息帧传输,造成该消息帧的延迟。

(1)仲裁延迟

由于消息之间碰撞的随机性,故可以把队列延迟看作一个随机变量,用 t_m 表示。设 T_i 为第 i 类帧在网络中处理时间, N_i 为第 i 类帧的个数, i 为第 i 类帧的到达率;则根据里托(Little)定理 [39] 有:

$$N_i = {}_i \times T_i \tag{3-15}$$

在队列延迟 tm 时间内,不考虑帧传输出错与重传,其 Ni 为:

$$N_{i} = \frac{t_{m}}{t_{i}} + \frac{t_{bit}}{t_{i}} \tag{3-16}$$

其中 t_i 代表的是这个消息帧两次传送时间间隔的最小值,即为帧 i 的传送周期。 t_{bit} 为在传输介质上传送一个数据位(bit)所需要的时间。若当前队列中有 n 个优先级比诊断帧 j 高的帧 j 则 j 帧的仲裁延迟为 :

$$T_{zc} = \sum_{i=1}^{n} N_i \times C_i = \left[\frac{t_m}{t_i} + \frac{t_{bit}}{t_i} \right] \times C_i$$
(3-17)

其中 C_i 为消息帧 i 的传输延迟,将在后面详细介绍。

(2)非仲裁延迟

若任意一帧消息在总线上的某一时间发送的概率密度为 $=1/T_{jo}$ 考虑 i 在 其占据总线后发送的情况,根据 M/G/1 优先排队论的 P-K(Pollaczek-Khinchin) 公式,在时间段 $[0,C_{i}]$ 中,当一个服务时间为 X 的新服务开始的时候,其残留

服务时间 r(t) (即在时刻 t 时,完成剩余服务所需要的时间)的初始值等于 X ,且在 r(t)个时间单位内呈线性衰减。我们可找到一个使 r(t)=0 的时刻 t0 ,得到非仲裁延迟的数学期望为:

$$T_{fzc} = \sum_{\forall i \neq j} \int_{0}^{C_{i}} r(t)dt = \sum_{\forall i \neq j} \int_{0}^{C_{i}} (C_{i} - t) \rho dt = \sum_{\forall i \neq j} \frac{C_{i}^{2}}{2T_{i}}$$
(3-18)

由于 Tzc 与 Tfzc 两部分相互独立,相加后可以得到队列延迟的数学期望:

$$t_{m} = \sum_{\forall i \neq j} \frac{\{(108 + x) \times 2\mu s\}^{2}}{2T_{i}} + \left[\frac{t_{m}}{t_{i}} + \frac{2\mu s}{t_{i}}\right]$$
(3-19)

2. 诊断报文的传输延迟

传输延迟指的是诊断消息从占据总线控制权到消息脱离总线的时刻。其延迟时间受到报文长度和总线上报文传输速度的影响。根据 CAN 2.0B 协议的规定,标准帧报文格式中最多包含 44 位(11 位标识符)的控制信息。而且,在数据帧中如果连续出现 5 个相同的位,就要在报文帧后插入一位填充位。而校验场界定符(1 位)、应答场(2 位)和帧结束(7 位)则不需要添加填充位。因此可以得知,标准帧的控制信息中有 34 位可能需要添加填充位。CAN 报文中的数据场可以装载 1 到 8 个字节的数据信息,所以可以计算得到在一个包含有 N 个字节数据的报文中,最多填充位个数为(N × 8+34—1)/5。设报文 m 的最长传输时间为 Cm,总线上传输一位所需要时间为 Thit,则标准帧的总传输时间为式 3-20。



图 3-5 标准帧的帧长度

$$C_t = \{108 + x\}T_{bit}$$
 (3-20)

 C_t 表示每帧的传输延迟,108 表示消息场仲裁场、控制场等固定格式部分长度。x 表示为 CAN 帧中因填充位所增加的位长度(0 < x < 8), T_{bit} 表示位时间。由于诊断系统采用高速 CAN 进行传递,波特率为 500K/s 根据式(3-21)可得:

$$T_{bit} = \frac{1}{\cancel{b} + \cancel{b}} = \frac{1}{500K/s} = 2 \times 10^{-6} \, s = 2 \, \mu s \tag{3-21}$$

3.4 仿真实验

为了验证上面提出的 STmin 延迟的数学模型,以及在改变 STmin 时对整个网络的影响。根据参考文献^[28]提出的方法,我们在 CANoe 下建立如图 3-7 所示的诵信仿真平台功能为:

- (1)模拟现实汽车网络中的拓扑结构;
- (2) 模拟 CAN 总线的参数;
- (3) 记录所有 CAN 总线上所传输的消息帧的时间参数。

整个网络由 2 个部分组成,即:通过 CANoe 虚拟网络上的真实的 ECU 节点和其他节点。

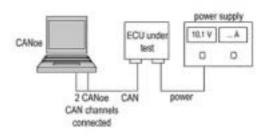


图 3-6 实验仿真平台

仿真系统模拟车身中舒适 CAN 消息传递系统,通过 CANoe 模拟 5 个节点(如表 3-3 所示),并通过每个节点向总线上发送网络管理消息(周期性消息)和应用层消息(事件性消息),在不同网络负载率的条件下(10%、20%、30%)分别比较固定的 STmin 参数和通过动态算法获得的 STmin 参数对消息帧的传输延迟。

节点	缩写	标识符
汽车座椅节点	DSCU	0x1F0
左门控制节点	DDCU	0x1E6
右门控制节点	PDCU	0x1EA
点火控制节点	PEPS	0x1D0
网关	GATEWAY	0x1F6
网络管理节点	NM	0x482

表 3-3 实验中的节点

(1) 在 30% 网络负载率下的周期性的消息帧进行进行延迟和碰撞实验。各个节点依次发送 200 帧,在此之中观察实际节点 DSCU 中的消息延迟和碰撞次数,如图 3-8 所示。

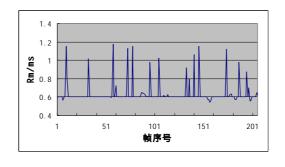


图 3-7 DDCU 在 30% 网络负载率下的时间延迟(只有周期性消息)

图 3-7 就是网络的负载率在 30%的时候 DDCU 帧消息延迟 , 其中图形上的尖峰就是由于碰撞所产生的延迟 , 由图 3-7 可以看出在 30%的时候尖峰数有 16 个 , 平均的消息延迟为 0.623。同理在网络负载率分别为 10%和 20%的情况下得到结果如表 3-4 所示。

	平均消息延迟	尖峰数	
10	0.602	3	
20	0.614	7	
30	0.623	16	

表 3-4 诊断网络中周期性消息实验结果

由上表可以发现随着负载率的不断提高平均的网络延迟和尖峰数也在不断的上升,这说明随着负载率的提高 DDCU 所发出的消息帧同其他消息帧的碰撞的次数增加。而平均消息的延迟的增加是因为 DDCU 消息的优先级较低,随着仲裁延迟的增加平均消息的延迟相应的加大。

(2) 在步骤(1)的基础上引入应用层消息(事件性消息),各个节点依次 发送 200 帧,在此之中观察实际节点 DSCU 中的消息延迟和碰撞次数,如图 3-8 所示。

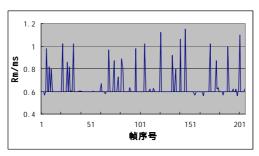


图 3-8 DDCU 在 30% 网络负载率下的时间延迟(有事件性消息)

同样在网络负载率分别为 10%和 20%的情况下得到消息延迟结果如表 3-5 所示。

 网络负载率/%
 平均消息延迟 Rm/ms
 尖峰数

 10
 0.612
 7

 20
 0.624
 16

 30
 0.643
 24

表 3-5 诊断网络中非周期性消息实验结果

同样由上表可以发现随着负载率的不断提高平均的网络延迟和尖峰数也在不断的上升,这说明随着负载率的提高 DDCU 所发出的消息帧网络上其他消息帧的碰撞的次数增加。然而平均消息延迟和尖峰数的增加是由于高优先级的事

件性消息的引入,使得 DDCU 消息在队列中的排队等待时间和仲裁时间的增加,这种情况对于车辆的控制是不利的。

(3) 动态改变了 STmin 参数后,分别在不同的网络负载率下进行仿真实验,得到的通信延迟结果如表 3-6 所示。

表 3-6 动态改变 STmin 参数后诊断网络中周期性消息实验结果

网络负载率/%	平均消息延迟	尖峰数
	Rm/ms	
10	0.590	0
20	0.592	1
30	0.594	2

可以看出不同的网络负载率的情况下周期性的消息延迟时间很稳定。平均的消息延迟和尖峰数都有所下降。

(4)事件性消息在不同网络负载率的情况下进行的延迟实验,结果如表 3-7 所示。

表 3-7 动态改变 STmin 参数后诊断网络中非周期性消息实验结果

网络负载率/%	平均消息延迟 Rm/ms	尖峰数
10	0.602	5
20	0.622	13
30	0.639	20

从上表可以发现在包含事件性消息的通信中平均消息的延迟和碰撞数都有 所下降但都不明显。

第四章 ISO 15765 应用层的建模

针对在较短的时间内根据不同需求如何选取诊断服务的问题,通过对 ISO 15765 诊断协议的应用层内容以及车载网络诊断系统应用层的特点详细分析后,利用有色 Petri 网作为工具并同诊断系统流程相结合,建立故障诊断的有色 Petri 网模型,描述诊断仪与 ECU 通信之间的逻辑关系。最后以诊断中的 FLASH 下载为例,建立了有色 Petri 网的诊断模型。通过仿真分析验证了模型的正确性和有效性且有助于提高车载网络诊断系统软件的开发效率。

4.1 ISO 15765 应用层分析

ISO 15765 中的应用层主要的功能有下面 2 个:

- (1)为用户提供响应诊断服务;
- (2)对所提供的服务进行时间的管理。

4.1.1 应用层诊断服务

ISO 15765 的应用层是在 ISO 14229 所提供的服务的基础上,根据 CAN 总线的特征而制定的。它主要是规定服务的命令格式,但是没有规定服务的具体内容。这就使得在协议的框架下,用户需要根据自己的需求选定和设计相关的服务。

本研究将整个的应用层协议分为如下几块:通信管理、权限管理、监控管理、FLASH 编程、其他功能控制。其中每个小模块又会包含多个子诊断服务命令,见表 4-1。

1.通信管理

在 ISO 16765 应用层协议中为了使通信能够顺利的进行,因此在通信开始的时候需要设定开始通信、通信结束以及通信的测试时连接状态和管理信息。通过这些服务我们可以开始进行诊断服务、结束诊断服务、更改超时信息等信息的设置。

2. 权限设置

针对不同的客户,如不同的设备供应商、整机制造商、电控单元制造商、整车制造商和维修厂家等,ISO 15765 提供众多的诊断服务和不同的诊断信息。 在不同的权限下,能够访问不同的的电控单元储存器资源,读取不同的故障、相关信息及能够进行不同的测试服务。

3. 监控模块

监控模块主要包含对电控单元的信息读取,以及当获取电控单元的一些实时信息(如当前的车速、档位等)统计数据(发动机的运行时间、整车的运行距离等)在出场前写入电控单元的具体信息(出场日期、生产批号等),以及

在车辆行驶时候故障信息。

0:表示不需要通过安全访问 1:表示需要通过安全访问 服务号 诊断服务名称 默认 编程 扩展 28 CommunicationControl 0 83 AccessTimingParameter 0 通信管理 3E TesterPresent 0 0 0 85 ControlDTCSetting 0 10 Start Diagnotic Session 0 0 0 权限管理 27 AccessTimingParameter 0 0 22 ReadDataByIdentifier 0 0 19 ReadDTCInformation 0 0 监控管理 2E WriteDataByIdentifier 1 14 ClearDiagnosticInformation 31 RoutineControl 1 34 RequestDownload 1 **FLASH** 36 TransferData 1 编程 37 RequestTransferExit EcuReset 0 0 11 其他功能

表 4-1 应用层服务分类

4. FLASH 编程

2F

在开发阶段,可以使用编程工具通过 BDM 接口来对 ECU 进行 Flash 编程。 然而在电控单元出厂后,需要灌装程序、电控单元下线时灌装最终数据和更新 MAP 数据这种方式就显得非常不适用。

1

InputOutputControlByIdentifier

ISO 15765 协议规定了可以使用故障诊断接口来进行 Flash 编程,通过诊断接口无论在 ECU 开发过程中、在生产线上、在整车上和在维修车间都可以进行电控单元的数据和程序更新。

5. 其他的功能

除了上述的功能以外,还有一些零散的功能,如 Ecu Reset 功能用于对系统进行软复位,以及在下载完成后的硬启动。在电控单元接收到改条命令后,检查复位条件如果在满足条件后利用看门狗实现软件的复位。

InputOutputControlByIdentifier则是用来触发一些预订的测试任务来获取整车或者整车部件或者子系统的状态,如座椅的位置调整可以测试出座椅继电器的工作状态。

4.1.2 应用层时间管理

同样应用层的时间管理是为了保证发送端和接收端不会因为等待而永久的

挂起,从而进入死循环。因此在每次服务的工作过程中设置了超时的时间参数,如图 4-4 所示。

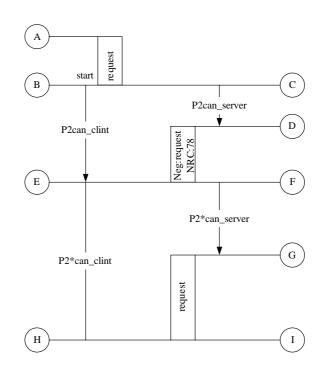


图 4-1 应用层时间控制

上图是一个简单的默认会话状态下点对点的通信过程。

A:开始一个新的诊断服务。

B: 当诊断仪在发送一个新的诊断服务后启动定时系统 P2can_clint。

C: 当电控单元接收到了诊断服务请求后启动定时器 P2can server。

D: 当电控单元在规定时间内无法按时完成请求服务时发送延迟的消极消息报文。

E:在接收到消极延迟消息报文后诊断仪停止 P2can_clint 定时器、启动新的定时器 P2*can_clint。

F:在发送完延迟消极消息报文后电控启动定时器 P2*can_server。

G:电控单元完成服务请求,根据结果发送响应报文。

H:诊断仪接收到响应报文停止计时器 P2*can clint。

I:发送完成默认会话结束。

4.2 Petri 网理论

Petri 网是由 Carl Adam Petri 发明的。最初只有自动化专家对 Petri 网感兴趣,后来 Petri 在计算机科学中也得到了广泛的应用,例如,在性能评价、操作系统和软件工程等领域,Petri 都得到了比较广泛的应用。特别是已经证明,用Petri 网可以有效的描述并发活动。

Petri 网包含 4 种元素:一组位置 P、一组转换 T、输入函数 I 以及输出函数 O,图 4-2 举例说明了 Petri 的组成。

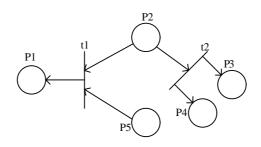


图 4-2 Petri 网的组成

其中,位置 P 为{P1、P2、P3、P4、P5},在图中用圆圈代表位置;转换 T 为{T1、T2},在图中用短直线表示转换;两个用于转换的输入函数,用由位置指向转换的箭头表示,它们是 I $\{t1\}$ ={P2,P4}和 I $\{t2\}$ ={P2};两个用于转换的输出函数,用由转换指向位置的箭头表示,它们是 O(t1)={P1}和 O(t2)={P3,P4}。

所以更形式化的 Petri 网结构,是一个四元组 $C=\{P, T, I, O\}$ 。 其中:

P={P1,···, Pn}是一个有穷位置集, N>0;

 $T=\{t1, \dots tm\}$ 是一个有穷的转换集合, M>0, 且 T 和 P 不相交;

I:T->P 为输入函数,是由转换到位置无序单位组的映射;

O:T->P 为输出函数,是由位置到转换无序单位组的映射。

Petri 网的标记是在 Petri 网中权标记托肯(token)的分配。例如,在图 4-2 中有 4 个托肯,其中有一个在 P1 中,两个在 P2 中,P3 没有,还有一个在 P4 中。上述标记可以用向量(1,2,0,0,1)表示。由于 P2 和 P4 中有权标,因此 T1 启动。通常当每个输入位置所拥有的权标数大于等于从该位置到转换的线数大于等于从该位置到转换的线数时,就允许转换。被激发时,P2 和 P4 上各有一个权标被移除,而 P1 上则只增加一个权标。Petri 网中权标总数不是固定的,在这个例子中两个权标被移除,而 P1 上只能增加一个权标。

在图 4-2 中 P2 上有权标,因此 T2 也可以被激发。当 T2 被激发时,P2 上将移走一个权标,而 P3 和 P4 上各增加一个权标。Petri 网具有非确定性,也就是说,如果数个转换都达到了激发的条件,则其中任意一个都可以被激发。图 4-2 所示 Petri 网的标记为(1,2,0,0,1), T1 和 T2 可以同时触发。假设两者同时触发则结果如图 4-3 所示,标记为(2,0,1,1,0)。

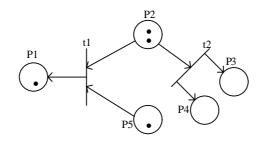


图 4-3 未激发的 Petri 网

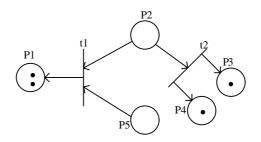


图 4-4 激发后的 Petri 网

更形象化的说,Petri 网 C=(P,T,I,O) 中的标记 M,是由一组位置 P 到一组非负整数的映射:

 $M: P \longrightarrow \{0, 1, 2, \dots\}$

这样,带有标记的 Petri 网成为一个五元组{P, T, I, O, M}。

4.3 诊断服务 Petri 网建模

4.3.1 车载网络诊断系统的结构和特点

在总线式的车身诊断控制系统中,分布与汽车车身的多个电子控制单元 (ECU)是通过 CAN 总线连成网络,而每个电子控制单元都包含了与之相应 的诊断服务功能。通常在诊断的时候选择诊断仪为中央控制节点,并以逻辑的 方式实现对车身中各个 ECU 节点的控制和诊断。

总线式车载网络诊断系统的工作流程为:诊断仪改变会话状态; ECU 发送响应消息;诊断仪发送所需进行的诊断服务;相关的 ECU 接收到命令消息,并根据命令作出相应的动作后按照确定的规则产生相应的消息并回复。

由于诊断服务原语的离散性,其状态组合构成了整个车身控制系统的状态空间,而 ECU 内各个服务原语的状态改变、响应消息的接收可以作为事件。在报文消息的触发下,诊断报文响应消息按照一定规则发生状态变化,由此可见车载网络诊断系统有着离散时间系统的典型特征:

- (1)状态的变化遵循有一定的逻辑规则;
- (2)状态的变化是由事件驱动,并只在离散时间点发生改变。

4.3.2 车载网络诊断系统的 Petri 网建模方法

如果只是使用基本的 Petri 网来为车载网络诊断系统, ECU 中每个服务的状态只能有一种状态。而选择 CPN 来建模,则可以将每个器件用一个库所描述,在通过不同的颜色的托肯来表示其不同的状态。

诊断仪任何的一个请求服务的变化都会触发产生响应的 ECU 的响应状态消息,这一事件可以用一个变迁来描述,其发送的消息可以用一个普通的托肯表示。响应 ECU 接收到请求服务消息后,会检测与之相关的服务状态,然后根据规则产生响应的响应消息,这个过程对应的是一个变迁。产生的响应消息可以让控制单元改变状态或者读出该单元目前的状态,因而可以用一个库所表示。控制单元的响应消息也是一事件,需要使用一个变迁来表示。

每个服务在任何一个时刻只会处于一种状态,因此在表示服务的库所之中任何时刻只存在着一个有色托肯。对于表示诊断仪的库所在同一时刻只能发送一种服务命令消息因此其库所容量也要限为 1,由此我们对有色 Petri 网添加了如下的约定:

约定 1 在 中,各个库所包含的托肯最多只能为 1,既 \forall $p \in P$: M(p) 1。

而车载网络诊断系统包括通信管理、权限管理、监控管理、FLASH编程等子模块。此时可以通过建立各个子模块的 CPN 后再合并各个子网模型中的相同的库所和变迁然后合并成整个 ECU 的车载网络诊断模块。由于篇幅所限,下面仅以 FLASH 编程模块为例子进行说明,并给出 CPN 的子模型。

4.3.3 车载网络诊断系统的 Petri 网子网模型举例

在车载网络诊断 FLASH 编程子模块中,请求会话层服务包含了默认会话、编程会话、扩展会话三个子请求消息。这三个会话分别是诊断仪请求所对应的 ECU 将会话模式调整到响应的会话方式,只有在响应的会话模式下才能进行响应的诊断服务。例程服务请求包含了擦除请求服务、下载控制服务,例程请求必须在编程会话模式下才能进行。安全访问请求服务包含了安全种子发送请求服务、安全码发送服务。其中安全种子发送服务必须完成之后才能进行安全码发送服务,此服务只有在编程会话下进行。再者 FLASH 下载服务中包含了请求下载服务、下载服务、和退出下载服务。只有在完成了安全访问请求服务和路径请求服务后才能进行 FLASH 下载服务。

由此可见诊断仪发送的服务请求都是输出器件,而 ECU 返回的响应消息都是输出器件。我们根据上述的描述将库所定义如下:

P1:默认会话回复状态:

P2: ECU 会话层状态;

P3:扩展会话回复状态;

P4:编程会话回复状态;

P5:安全访问服务码回复状态;

P6:安全算法通过状态;

P7:清除例程完成状态:

P8:安全访问回复状态;

P9:下载状态控制状态;

P10: 例程服务回复状态;

P11:请求下载服务状态;

P12:下载服务状态;

P13:退出下载服务状态。

为了方便理解我们将 CPN 模型划分为三层:

- (1) 状态转换层:完成服务的状态转换并的到正确的响应。
- (2)下载前的判定层:车身的 flash 下载服务必须在获得相应的权限后才能进行。因此在进入了相应的会话状态后,必须判断此时是否获得权限和是否完成了下载前的准备。只有在获得了此层的正确响应后才能继续进行下一层的服务。
- (3) Flash 下载层:此层主要是接收下载的数据,并判断数据是否符合规范,最后给出最终的结果。

整个的状态转换,如下图 4-5 所示。

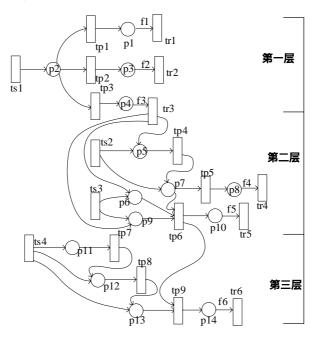


图 4-5 Flash 下载 Petri 网建模

图中变迁代表的含义如表 4-2 所示。

表 4-2 变迁含义

变迁	含义
ts1	服务状态请求发生变化
ts2	安全访问服务请求发生变化
ts3	例程服务请求发生变化
ts4	flash 下载服务请求发生变化
tp1	默认会话状态发生改变
tp2	扩展服务状态发生改变
tp3	编程服务状态发生改变
tp4	安全访问服务码状态发生改变
tp5	安全访问许可发生改变
tp6	例程服务状态发生改变
tp7	下载请求服务状态发生改变
tp8	下载进行服务状态发生改变
tp9	下载结束服务状态发生改变
tr1	产生默认会话回复消息
tr2	产生扩展会话回复消息
tr3	产生编程会话回复消息
tr4	产生安全服务完成回复消息
tr5	产生例程服务完成回复消息
tr6	产生 flash 服务完成回复消息

对于上述的库所,将其颜色定义如下:

- C(P1)=(无消息,默认会话通过,默认会话否决);
- C(P3)=(无消息,扩展会话通过,扩展会话否决);
- C(P4)=(无消息,编程会话通过,编程会话否决);
- C(P5)=(安全码访问请求正确、子服务号不支持、服务请求长度错误);
- C(P6)=(安全访问请求正确、安全码不正确、服务请求长度错误);
- C(P7) = (擦除例程通过,擦除例程不通过);
- C(P8)=(无消息,下载控制例程通过,下载控制例程不通过);
- C(P9)=(安全服务正确回复,安全访问顺序不正确,服务长度请求错误、服务会话状态不正确);
- C(P10) = (无消息,例程服务正确回复,服务长度请求错误、服务会话状态不正确);
 - C(P11)=(请求下载服务通过、请求下载服务不通过);
 - C(P12)=(下载服务通过、下载服务不通过);
 - C(P13)=(退出下载服务通过、退出下载服务不通过);
- C(P14)=(无消息,flash下载服务正确回复,服务长度请求错误、服务会话状态不正确、安全访问未通过状态);

在此颜色定义中我们将整个消息的分为积极响应消息、消极响应消息。

约定 2 在 中,当库所获得特定颜色的托肯时,立刻触发变迁发送消极相应消息相应托肯消失。

由此一来我们可以将最终输出的函数表达式定义如下:

上述模型中的消极响应消息我们赋予了不同颜色的托肯表示不同类的消极响应消息。当需要发送消极消息时,系统根据不同颜色的托肯自动发送相应的消息消息。

对于车载网络诊断系统的其他子系统,也都是采用这种统一的形式对相应的库所和变迁进行划分建立 CPN 子网模型。

4.4 仿真实验

在车载网络诊断控制系统的 CPN 模型中,变迁 tsi 表示着诊断仪(输入器件)发送了新的诊断服务消息,而变迁 tri 表示相应的 ECU(输出器件)根据输入的消息返回的响应服务消息。所以我们可以根据诊断仪发送的服务消息产生的变迁发生序列,实现对车载网络诊断系统的仿真。对于上述的变迁我们用(x,Y(x))表示。

我们仍以 FLASH 编程子模块为例。假设首先诊断仪发出编程会话服务请求,在收到积极响应后发送安全服务码服务请求。根据返回的服务码计算得到最后的安全码并发送消息获得相应的访问权后发送例程服务,最后完成 flash下载服务。其 CPN 的变迁序列有可能为:

(ts1,(默认会话、编程会话、扩展会话));

(tp3,扩展会话)—> (tr3,扩展会话通过);

(ts2, (安全码访问、安全码计算服务))—> (tp4, 安全码访问请求;)

(ts2, (安全码访问、安全码计算服务));

(tp5, 安全码计算服务) -> (tr4,安全服务通过);

(ts3,(擦除例程服务、下载例程控制服务));

(tp6, 擦除例程服务) —> (tr5, 擦除例程服务通过);

- (ts4,(请求下载服务、下载服务、退出下载服务))—> (tp7,请求下载服务通过);
- (ts4,(请求下载服务、下载服务、退出下载服务))—> (tp8,下载服务通过);
 - (ts4,(请求下载服务、下载服务、退出下载服务));
- (tp9, 请求下载服务通过、下载服务通过、退出下载服务通过)—> (tr6, flash 下载服务正确)。

可以看出,通过仿真诊断仪按照流程发送了相关的消息命令后,根据 Petri 网所建立的模型, ECU 也发出了下载成功的相应消息,表明此模型是有效、可行的。

第五章 ISO 15765 实现及测试

在对协议的网络层和应用层进行了深入的分析后,选取了座椅电控单元诊断系统作为实现及测试目标。通过对电子控制单元诊断系统的架构设计、软件开发环境以及设计思路行了简单的说明,最后通过 CANoe 总线仿真工具对所做的座椅电控单元诊断系统进行验证与分析。

5.1 座椅电控单元软件构架

诊断控制单元的软件架构设计是软件代码实现的前提,架构设计的好坏 将直接影响到代码和功能是否能够顺利的实现。ECU 的架构按照功能来分的 话可以分为以下几个部分:

- (1)应用层软件驱动;
- (2) CAN 设备驱动;
- (3)信号交换层处理;
- (4) CAN 通信配置管理;
- (5) 网络管理;
- (6)传输层管理;
- (7)诊断服务层;
- (8)诊断应用层;
- (9)应用层;
- (10) Boot Loader 层。

整个的软件分层构架如图 5-1 所示[42]:

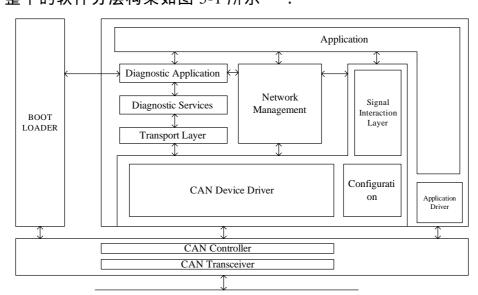


图 5-1 诊断控制单元软件分层构架

5.2 软件开发基层环境

5.2.1 开发环境

诊断控制单元的开发系统我们用的是 μ C/OS-II 操作系统, μ C/OS-II 是一个免费的、源代码公开的嵌入式实时多任务内核,是专门为嵌入式应用设计的 RTOS,提供了实时系统所需要的基本功能。 μ C/OS-II 的全部功能的核心部分代码只占用 8.3KB,用户还可以针对自己的实际系统对 μ C/OS-II 进行裁剪。 μ C/OS-II 只提供了诸如任务调度、任务管理、时间管理、内存管理、中断管理和任务间的同步与通信等实时内核的基本功能。

 $\mu C/OS$ -II 已有十余年的发展历史,在诸多领域得到了广泛应用。许多行业中 $\mu C/OS$ -II 成功应用的实例,也进一步说明了该内核的实用性和可靠性。其特点为:

- (1)可移植性;
- (2)可裁剪;
- (3)占先式;
- (4) 多任务。

5.2.2 开发工具

本项目所使用的 MCU 是 Freescale 公司的 MC9S08DZ60 芯片 ,集成开发环境(以下简称 IDE) 采用的是 Metrowerks 公司开发的 Codewarrior 软件。飞思卡尔所有系列的微控制器都可以在 Codewarrior IDE 下进行软件开发 ,对应的软件集成开发环境采用的是 Code Warrior ,在此环境中可以配置和具体 MCU 对应的所有设置 , 然后就进行代码的编写了 , 最后再对代码进行编译。

目前 Codewarrior IDE 在汽车电子软件开发领域是用得比较普遍的集成开发环境,其有以下几个主要的优点:

- (1) 交叉平台开发;
- (2)支持多种语言;
- (3) 开发环境接口统一;
- (4)支持插件工具。

5.3 诊断系统设计

从图 5-1 可以看出,整个诊断控制单元设计分为如下几个部分: CAN 驱动层、诊断系统设计、网络管理设计、Boot Loader 设计,电控单元内部空间按图 5-2 所示分配。

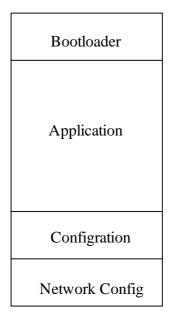


图 5-2 内部空间的 MEMORY

从图 5-2 可见,按照内存地址的顺序依次分配给引导区、应用层数据区、 电控单元数据标定区、网络配置区。

5.3.1 引导区设计

作为程序最先执行的区域,引导区的主要职责是系统自检以及基于 CAN 的功能升级和完成内存映像。

5.3.2 应用层数据区设计

应用层数据区数据区主要包括 CAN 的内部资源、应用层设计、诊断层设计和网络管理设计。

CAN 的内部资源主要包括了 CAN 收发器、CAN 管理器、CAN 的设备驱动,良好的 CAN 底层驱动能够增强电子控制单元的软件质量。这部分的工作主要是对内部所有寄存器的初始化和接收、写入相应的控制指令,为了保证程序在复杂车况下运行的可靠性,对软件的这部分代码具有极高的要求,必须保证驱动的质量。

应用层设计主要是完成电控单元应该完成的功能性操作。对于座椅控制电 控单元来说包含了何时启动电控单元、当接收到了何种消息时候需作出何种动 作等。

在诊断层方面,在结合整车厂的功能要求后车身控制的诊断代码遵循 ISO 15031-6 标准进行开发。汽车电子 ECU 的开发方面,在一般情况下不但需要能够完成 ECU 的指定功能外,还需要进行诊断、网络管理等功能的完成。目前几乎所有国外的主机厂以及国内的主机厂都要求零部件供应商在提供的 ECU 软

件单元同时加入诊断功能,通过车载诊断接口(OBD, on boarddiagnostic)能够迅速的读出汽车上何处的 ECU 出现了问题,从而能为汽车电控模块的错误定位和后续的维修提供了极大的便利。

车载诊断部分的功能我们按以下的方法进行实现:假设车辆行驶正常时,各 ECU 运行均正常,一旦某些设备在运行时出现了不正常的情况,诊断系统将会按照协议的具体规定将所对应的 DTC(Diagnostic Trouble Code,诊断错误码)记录下来并且存储到 EEPROM 里面。检测人员则可以利用相应的诊断仪通过诊断接口将 DTC 读出,根据错误发生的类型,协议规定将错误代码分为:历史性错误代码、间隔性错误代码以及当前错误代码,所有的错误代码按照诊断协议所指定有不同的格式进行显示。

网络管理层方面部分,本系统中由于时间的限制没有对这部分代码进行实 现。

5.3.3 电控单元数据标定区设计

主要是存放固定的数据, DTC 的编码、以及产品信息等数据。

5.4 诊断系统的测试

5.4.1 测试环境介绍

对于所设计的诊断单元我们通过 CANoe 测试工具进行测试。CANoe 是由德国 vector 公司开发的具有网络监测功能、分析功能和强大的系统仿真功能的系统级总线开发工具^[46]。同时我们可以通过 Vector 的 CAN 总线接口实现虚拟总线与真实物理总线的连接。

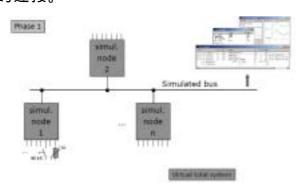


图 5-3 虚拟仿真系统

通过 CANoe 软件,我们可以实现 CAN 总线以全部虚拟节点进行的全数字仿真(如图 5-3)。此过程主要是针对具有具体数据的网络节点,通过 CAPL(CAN Access Programming Language)编程语言对节点报文的发送、接收、环境变量的变化或者是错误的出现等进行处理。将所设计的模拟系统通过 CANoe 进行离线仿真,检验各节点设计是否合理完善^[47-49]。

物理节点和虚拟节点相结合的半实物仿真(如图 5-4)也可以通过 CANoe 实现。进行控制器真实节点的开发,并将其与 CAN 总线相连接,同时使其它虚拟节点与虚拟总线保持良好连接,启动物理节点和虚拟节点的 CANoe 仿真环境,同时观察 CAN 总线的消息传输、错误处理等通信机制是否正常。如果整个系统能够正常工作,则认为总线系统功能得到了初步实现^[50]。如果我们并行开发节点总线,则各节点之间的开发步骤互不影响,此过程中,我们可以使用不同的方法获取环境变量,实现真实总线环境的模拟。

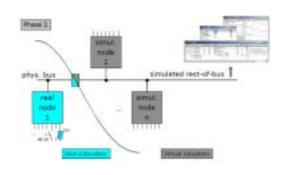


图 5-4 半实物仿真

对于真实物理节点的通信状况,我们也可以使用 CANoe 进行实时监控(如图 5-5)。CANoe 在此过程中可以检查到系统中各功能节点的详细情况,并作出分析^[46]。为了减少错误的发生,我们可以使用功能模块取代真实的网络节点,并通过两种状态之间的转换,实现检查功能。

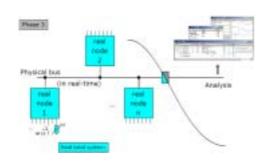


图 5-5 真实物理总线仿真

CAN 总线开发工具 CANoe 具有功能强大、操作简单等特点,可以对总线进行测试、分析、仿真和记录,它主要有以下几个部分组成^[46,50]:主窗口、仿真模型建立界面(Simulation Setup)、数据库 (CANdb++ Editor)、CAPL 编程浏览器(CAPL Browser)、程序生成器(CAPLGenerator)) 和面板编辑器(Vector Panel Designer),使用界面如图 5-6 所示。



图 5-6 CANoe 使用界面

5.4.2 故障诊断测试

为了进一步验证所制定的协议和所建立的汽车总线拓扑控制系统的合理性与规范性,本章对汽车总线网络系统进行半实物仿真,图 5-7 为在 CANoe 中进行半实物仿真的结构图。

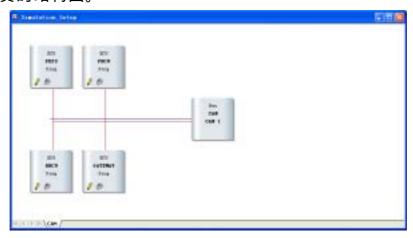


图 5-7 半实物仿真图

在半实物模型中我们模拟了和我们所设计的座椅节点相关其他 4 个节点: 左车门控制节点、右车门控制节点、网关节点并模拟了诊断仪的同实物座椅电 控节点的诊断过程。我们将整个测试部分分为两个部分:正向测试和反向测试。

1.正向测试

正向测试是指诊断仪按照预订的步骤进行,发送的报文完全按照规范要求。 其测试用途是查看所做的实物电控系统是否完成了需要完成规范要求。

对于正向测试我们运用的 CANoe 自带的 Diagnostic Console 测试工具其界面如图 5-8 所示。

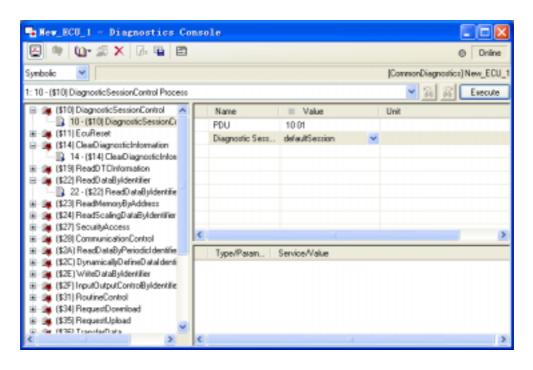


图 5-8 Diagnostic Console 测试工具

整个的测试步骤如下:

(1) 配置 Diagnostic Console 测试工具的基本属性,主要属性有:发送、接收诊断报文的 ID 号、应用层交互的时间设定(S3_Server、P2_Server 等)。

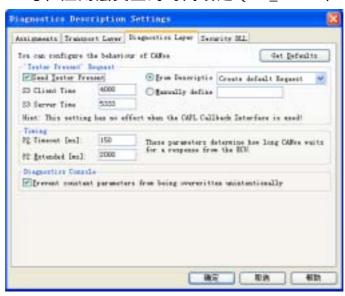


图 5-9 Diagnostic Console 测试工具参数设置

- (2) 选取所需测试的服务号,设置所需发送的服务子服务内容数据并且发送。
- (3) 当测试结束之后会在测试结果显示区域显示电控单元返回的测试结果。

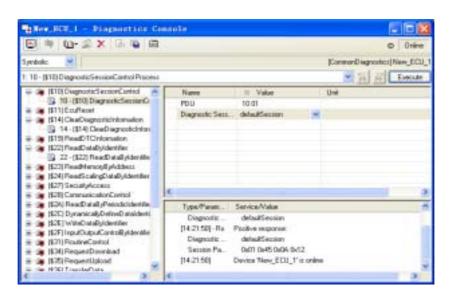


图 5-10 正确的测试结果

2. 反向测试

反向测试是指诊断仪不按照预订的步骤进行,或者发送的报文有违规范要求。其测试用途是查看所做的电控系统是否能按照规范要求发送正确的消极响应报文。

我们设计了如下几项错误进行测试。

错误编 错误类型 码 serviceNotSupported 0x11 0x12 subFunctionNotSupported incorrectMessageLengthOrInvalid 0x13 Format conditionsNotCorrect 0x22 0x24 requestSequenceError 0x31 requestOutOfRange 0x33 securi tyAccessDeni ed 0x35 invalidKey

表 5-1 错误类型设置

对于某些可以用 Diagnostic Console 测试工具直接测试的错误 ,我们直接对数据进行设置后发送错误数据 , 看是否能够返回我们所希望的错误消极报文。

如图所示在测试 28 号服务的时候我们将请求数据设置为规范所不支持的 子功能服务号后发送,得到了消极响应报文。

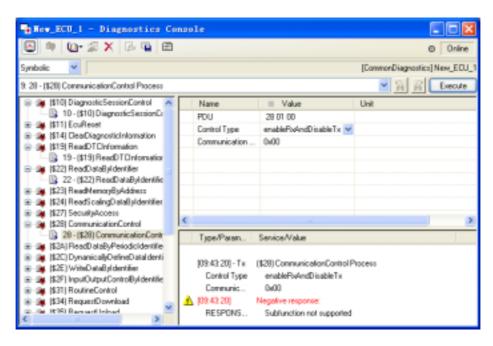


图 5-11 反向错误测试

对于不能用 Diagnostic Console 所进行的测试 ,则通过我们模拟的诊断仪节点进行测试。

第六章 总结和展望

6.1 本文研究主要工作的总结

随着汽车技术的日趋完善,以及 CAN 总线技术在车载网络中的成功运用,越来越多的电控单元(ECU)加入到了汽车车载网络中。针对如何选取合适的诊断协议对网络中的 ECU 进行控制,以及如何解决车身电控单元诊断系统软件开发过程中遇到了研发周期短、需求的变更较多等挑战。本文在以 ISO 15765 诊断协议为背景进行研究,所做工作如下:

- (1)研究了当前主流的可用于车身诊断系统的协议——SAE J1939 协议, ISO 14230 协议和 ISO 15765 协议。分析了各协议的网络层、应用层、应用领域以及各个协议的特点,通过分析比较了各个诊断协议的优缺点,选择将 ISO 15765 协议为研究的重点。
- (2)对 ISO 15765 诊断协议进行了详细的分析,并以网络负载率、数据帧优先级参数为约束,在对通信中的队列延迟和传输延迟分析的基础上,建立了 STmin 参数模型,并给出了最小平均时延的计算公式。通过模型的建立可以增加数据帧传输的稳定性并减少传输延迟。
- (3)将有色 Petri 网和诊断系统的流程相结合,建立故障诊断的有色 Petri 网模型。利用有色 Petri 网作为工具,描述诊断仪与 ECU 通信之间的逻辑关系。并以诊断中的 FLASH 下载为例,建立了有色 Petri 网的诊断模型。有色 Petri 网的引入可以有效的提高车载网络诊断系统的软件的开发效率。
- (4)针对 ISO 15765 诊断协议的研究,设计了车身座椅电控单元诊断系统,利用汽车总线设计开发工具 CANoe 搭建仿真环境,对所设计的电控单元诊断系统进行了仿真和验证实验,仿真结果表明此系统的可行性和可靠性。

6.2 展望

本文针对 ISO 15765 汽车诊断协议做了较深入的研究,但由于时间等因素的限制,还存在不完善的地方,下面是我们对下一步研究工作的展望:

- (1) 网络层建立的 STmin 参数延迟模型分析诊断协议的通信系统中不同负载率下的数据帧的延迟时间。虽然此模型虽然可以有效的减少周期型消息帧的延迟时间,但是其对减少事件型消息帧延迟时间的效果是有限的。如何减少事件型消息帧的延迟时间也是下一步工作的一个方向。
- (2) ISO 15765 诊断协议目前应用与 CAN 网络中。而如何根据协议的规定制定可用于其他总线的通用诊断协议和网关也是下一步的工作。

参考文献

- [1] 魏培敏.浅谈制约我国汽车电子产品发展的因素[EB/OL].http://www.q cdz.cn/html/doc/20089/54474.html,2009,03,10
- [2] 陶侃.基于 SAE J1939 协议的商用车信息控制单元设计[D]. 武汉:华中科技大学,2009.
- [3] 诸晓强.基于 CAN 总线的车身控制系统的设计与实现[D]. 上海:上海 交通大学,2010.
- [4] International Organization for Standards. ISO 15765-2: Rord Vehicl es-Diagnostics on Controller Area Networks(CAN)[S].2004.
- [5] International Organization for Standards. ISO 15765-3: Diagnosetic s on Controller Area Networks(CAN)[S]. 2004.
- [6] LIU SH,RAJKUMAR,SATHAGESS. Generalized rate-monotonic scheduling: a framework for developing real-time system [J].IEEE,1 994,82(1):68-81.
- [7] Society of Automotive Engineers. SAE J1939-01 Recommended Pr actice for Control and Communication Network for On-Highway E quipment[S].2000.
- [8] Society of Automotive Engineers. SAE J1939-11 Physical Layer, 250Kbit, Twisted Shielded Pair[S]. 2006.
- [9] Society of Automotive Engineers. SAE J1939-13 Recommended Practice Off-Board Diagnostic Connector[S].2004.
- [10] Society of Automotive Engineers. SAE J1939-15 Recommended Practice Reduced Physical Layer, 250Kbits, Un-Shielded Twisted Pair [S].2003.
- [11] Society of Automotive Engineers. SAE J1939-21 Recommended Practice Data Link Layer[S].2006.
- [12] Society of Automotive Engineers. SAE J1939-31 Recommended Practice Network Layer[S].2004.
- [13] ISO14230 Road Vehicles Diagnostic Systems Keyword Protocol 2000- Part 3 Application Layer 2000 .
- [14] 饶运涛,邹继军,郑勇芸. 现场总线 CAN 原理与应用技术[M]. 北京:北京航空航天大学出版社,2003.
- [15] 麦克斯. 汽车第二代车载诊断系统解析[M]. 北京:机械工业出版 社,2007.
- [16] 任培,周经伦,罗鹏程等. 基于排队论的数据链系统信息传输时间延迟分析[J]. 计算机科学,2008,35(8):93-94,128.

- [17] 张宏,詹德凯,林长加. 基于 CAN 总线的汽车故障诊断系统研究与设计[J]. 汽车工程,2008,10 (30):934-937.
- [18] 丁志华,罗峰,孙泽昌. 基于 CANoe 的汽车故障诊断系统研制[J]. 汽车工程,2007, 29(5):449-452.
- [19] 史久根,张培仁等. CAN 现场总线系统设计技术[M]. 北京:北京航天航空大学出版社,2004.
- [20] 李佳,朱元等. CAN 与 TTCAN 通信延迟时间的分析[J]. 清华大学学报,2006,46(2):261-265.
- [21] 马向华,谢剑英. CAN 网络控制系统网络时延分析及其控制[J]. 化工自动化仪表,2006,33(1):51-55.
- [22] 袁崇义. Petri 网原理与应用[M]. 北京:电子工业出版社,2005.
- [23] Tadao Murata. Petri net sproperties analysis and applications[J]. Proc of the IEEE,1989,77(4):541-580.
- [24] Koriem S M. A Fuzzy Petri Net Tool for Modeling and Verification of Knowledge-Based Systems[J]. The Computer Journal, 2000, 43(3):206-223.
- [25] 刘小平,韩江洪. 基于有色 Petri 网的车身控制系统建模方法[J]. 汽车工程,2009,31(8),698-671.
- [26] 韩江洪,魏振春,张本宏. 总线式车身控制系统的规则化描建模方法[J]. 汽车工程,2006,28(12):1121-1124.
- [27] 陈玉宝,夏继强,邬学礼. Petri net 在故障诊断领域的应用研究[J]. 中国机械工程,2002,11(12):1386-1388.
- [28] 杨其宇,张霞. 模糊 Petri 网故障诊断技术应用[J]. 测控技术,2006,25 (4):73-74.
- [29] 王俊国,闽松,赵金等. 基于模糊 Petri 网的故障诊断方法[J]. 兵工自动化,2002,21(6):18-21.
- [30] 李明波. 基于 CAN 总线的实时通讯研究[D]. 北京:北京化工大学,20-06.
- [31] 宋君花,唐航波,熬国强等. 混合动力汽车整车控制器硬件电路与CAN 总线通信的设计开发[J]. 汽车工程,2007,29(7):590-593.
- [32] 张彤,朱磊,袁银南等. 基于 CAN 的混合动力汽车监控平台的开发[J]. 汽车工程,2008,30(6):496-500.
- [33] 汽车总线设计及测试宝典——汽车电子工程师必备. http://www.E-E Tchina.com,2009.
- [34] 李永强,宋希庚,薛冬新. CAN 局域网及 J1939 协议在货车和客车上的运用[J]. 汽车工程,2003,25(4):377-379.

- [35] 张恩勤. 汽车电子软件的趋势 AUTOSAR 如何改变您使用飞思卡尔控制器的方式[G]. 2007-11-28.
- [36] 杨国青. 基于模型驱动的汽车电子软件开发方法研究[D]. 浙江:浙江 大学,2006.
- [37] ARM home page. http://www.arm.com
- [38] 周立功. ARM 微控制器基础与实战[M]. 北京:北京航空航天大学出版社,2003.
- [39] Jean J.Labrosse. 嵌入式实时操作系统 μC/OS- [M]. 北京:北京航空 航天大学出版社,2003.
- [40] 李群芳,肖看. 单片机原理、接口及应用——嵌入式系统技术基础[M]. 北京:清华大学出版社,2005.
- [41] 吴刚. 基于 CAN 总线的 BCM 控制器设计研究[D]. 合肥:合肥工业大学,2008.
- [42] 卫星. 汽车车身 CAN 总线控制系统应用与研究[D]. 合肥:合肥工业大学,2005.
- [43] 李向明. 面向汽车电子的 CAN 网络应用层规范与协议开发[D]. 重庆: 重庆邮电大学,2006.
- [44] 陆文昌. 基于 CAN 总线的汽车发动机智能测控节点的设计[J]. 汽车工程,2003,25(5):463-466.
- [45] 邵梁强. 车身 CAN 总线系统的软件开发[D]. 南京:南京理工大学,20-07
- [46] 韩江洪,魏振春,张本宏等. 总线式车身控制系统的规则化建模方法[J]. 汽车工程,2006,28(12):1121-1124
- [47] Vector Informatik GmbH. CANoe User Guide. Eberdingen: Satzteam Fotosatz &Neue Medien Gmbh[G], 2003.
- [48] 张新波,孙泽昌,罗峰. 使用 CANoe 对车身控制器局域网络仿真的研究[J]. 江苏大学学报,2003,24(5):36-38.
- [49] 丁志华,罗峰,孙泽昌. 基于 CANoe 的汽车故障诊断系统研制[J]. 汽车工程,2007,29(5):449-452.
- [50] 张琛. 基于 DSP 的 CVT 控制系统的 CAN 总线研究[D]. 镇江:江苏大学,2009.
- [51] 宋艳红. 轿车电子稳定性控制系统的 CAN 通信研究[D]. 长春:吉林 大学,2008.
- [52] R. I. Davis, A.Burns, R. J. Bril, et al. Controller area network(c-an) schedulability analysis: Refuted, revisited and revised [J]. Real Time Systems, 2007, 35(3):239–272.

- [53] M. Short, M. Pont. Fault-tolerant time-triggered communication using CAN[J]. IEEE Transactions on Industrial Informatics,2007,3(2): 131–142.
- [54] Y. Tian, E. Ekici, F.Ozg"uner. Energy-constrained task mapping and scheduling in wireless sensor networks[J]. Proc. of the Workshop on Resource Provisioning and Management in Sensor Networks (RPMSN'05),2005,5:129-138.
- [55] Y. Yu, V. K. Prasanna. Energy-balanced task allocation for collaborative processing in wireless sensor networks[J]. ACM/Kluwer Journal of Mobile Networks and Applications, 2005, 10(4):115–131.
- [56] Nolte T,Hansson H,Bello L. Automotive communications-past,current and future[C]. 10th IEEE Conference on Emerging Technologies and Factory Automation,2005,8(1):8-10.

攻读硕士学位期间发表的论文

- [1]韩江洪,周涛,张本宏. 基于有色 Petri 网的车载网络诊断系统建模方法[J]. 合工大学报(已录用).
- [2]韩江洪,周涛. 汽车故障诊断协议中 STmin 的估值方法[J]. 汽车工程 (已投稿)

读研期间参与的科研工作

[1] 项目名称:基于 CAN 的车用网络系统集成协议及系统开发

项目背景:广东省教育部产学研项目(编号:2008B090500226)

[2] 项目名称:乘用车车身控制系统开发平台

项目背景:安徽省信息产业厅发展专项(编号:2008014)

[3] 项目名称:基于分层模型和规则描述的离散事件系统形式化方法

项目背景:国家自然科学基金资助项目(编号:60873003)

IS015765协议的研究与实现

作者: 周滔

学位授予单位: 合肥工业大学

□ 万万数据 WANFANG DATA 文献链接

本文读者也读过(3条)

- 1. 刘国权. 张伯英. 宋卫锋 KWP2000协议分析及开发测试[会议论文]-2006
- 2. <u>颜伏伍. 刘陶然. 胡杰. YAN Fuwu. LIU Taoran. HU Jie</u> <u>汽车故障诊断系统通信协议栈的研究与应用[期刊论文]—武汉理工大学学报(信息与管理工程版)</u> 2011, 33 (6)
- 3. <u>张丽. 方成. 李建秋. 黄海燕. ZHANG Li. FANG Cheng. LI Jian-qiu. HUANG Hai-yan</u> 基于ISO 15765的柴油机故障诊断接口开发[期刊论文]-车用发动机2009(4)

本文链接: http://d.g.wanfangdata.com.cn/Thesis_D143338.aspx