

# 基于 CAN/LIN 总线的实时操作系统的设计与实现

姚 勇, 涂时亮, 陈章龙

(复旦大学计算机科学与工程系, 上海 200433)

**摘要:** 论述了一个适用于 CAN 和 LIN 总线的实时操作系统 FDCX08 设计和实现, 介绍了该系统在一个通用控制器上的应用。FDCX08 主要用于 CAN 或者 LIN 或者 CAN 和 LIN 混合总线的微控制器上。根据 CAN 和 LIN 总线的特点, 设计了一个支持各个操作系统之间的任务通信的邮箱系统, 使得该系统可以被当作 CAN 节点、LIN 节点或者网关节点等各种节点。该系统具有实时性强、体积小、系统开销小、用户接口简单、便于移植等特点, 适用于汽车电子、工业控制等使用 CAN 和 LIN 总线的场合。

**关键词:** 实时操作系统; CAN 总线; LIN 总线; MCU

## Design and Implementation for Real-time Operating System Based on CAN and LIN Bus

YAO Yong, TU Shiliang, CHEN Zhanglong

(Department of Computer Science & Engineering, Fudan University, Shanghai 200433)

**【Abstract】** This paper describes the design and the implementation of an embedded real-time operating system (RTOS): FDCX08, used for CAN and LIN bus, and introduces the design of a usual-controller based on this operating system. FDCX08 is used in CAN, LIN or CAN and LIN mixed bus. According to CAN and LIN bus's character, it designs a mailbox system, which supports the communication of tasks among operating systems, to make FDCX08 as CAN node, LIN node, gateway node and so on. This system is real-time, lightweight, and simple-interface, which makes it suitable to be used in many fields, such as auto electron and industry control, where CAN and LIN bus are needed.

**【Key words】** Real-time operating system(RTOS); CAN-bus; LIN-bus; MCU

### 1 概述

作为工业总线最重要的网络技术之一, CAN 和 LIN 总线被广泛应用于工业控制、汽车电子以及机械电子领域, 为了更好地利用 CAN 和 LIN 总线, 我们设计了一个实时操作系统(RTOS)。CAN 和 LIN 总线一般用于 8 位机和 16 位机, 而对于操作系统来说, 8 位机对资源的要求更加严格, 所以, 我们以 8 位机为例来实现这个操作系统——FDCX08。

FDCX08 主要作为 MCU 用的操作系统。在整个系统中, 它可以作为 CAN 或者 LIN 总线的一个控制节点。图 1 为该操作系统应用的典型网络。

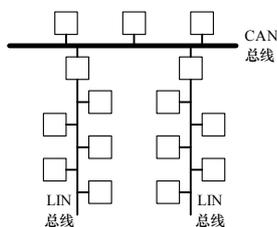


图 1 FDCX08 的典型应用

### 2 CAN 和 LIN 总线的基本概念及对操作系统的要求

#### 2.1 CAN 和 LIN 总线的基本概念

CAN-bus(Controller Area Network)即控制器局域网, 是一种面向一般工业设备的高速通信总线。它数据通信可靠, 每帧信息都有 CRC 校验及其它检错措施; 实时性高, 通信速率可达 1Mbps, 优先级高的数据最多可在 134ms 内得到传输; 数据精简灵活性强, 每帧数据量为 8 字节, 各个节点之间相

互独立, 总线上任何一个节点均可在任意时刻主动地向网络上其它节点发送信息。

LIN-bus(Local Interconnect Network)即本地互连网络, 是针对低成本应用而开发的汽车串行协议。它对现有 CAN 网络进行了补充, 支持车内的分层式网络, 价格低廉, 速度略低。

一般而言, 使用 CAN 和 LIN 总线的场合, 都是那些对实时性要求很高的工业控制、汽车电子领域, 而这些领域也非常重视成本。

#### 2.2 CAN 和 LIN 总线对操作系统的要求

基于 CAN 总线和 LIN 总线的特点, 操作系统需要满足以下的要求:

##### (1) 及时通信

CAN 和 LIN 总线作为一种通信类的总线系统, 对其操作系统的最主要的要求就是及时通信, 即任何的通信包都可以及时、正确地到达目的地。因此, 系统必须有一个强大的通信模块支持这种及时、正确的通信, 而且这个模块也必须符合 CAN 和 LIN 总线的特点。

##### (2) 硬件资源限制

CAN 和 LIN 总线系统必须在有限的内存空间和简单的内存管理单元 MCU 条件下可靠工作, 因此操作系统应尽量短小精悍。通用操作系统中常用的宏展开, 用在嵌入式系统中就不适合, 可以代之以函数调用, 以实现代码的复用。这

**作者简介:** 姚 勇(1980—), 男, 硕士生, 主研方向: 嵌入式系统; 涂时亮, 博导; 陈章龙, 教授

**收稿日期:** 2006-05-23 **E-mail:** yaoyong@fudan.edu.cn

是以牺牲一定的速度为代价的。只靠速度并不能保证实时性，事件必须在可预测的时间内得到处理，在最坏的情况下，实时性也应得到满足。能保证最坏情况下得到及时处理的设计都是合理的。

### (3)中断响应

中断响应是评价实时系统性能的重要标准。为了确保对关键数据的原子操作，屏蔽中断是必需的。但是在此过程中的中断屏蔽可能造成中断的丢失。同时，中断服务子程序阻塞了任务调度，影响了系统对关键任务的响应时间。因此，中断服务子程序应该尽快结束，而把进一步的处理交给上层来完成。

而对于 CAN、LIN 节点，为了达到通信的及时性，对于中断响应的要求，则必须保证通信及时无误地进行。

然而目前比较流行的嵌入式实时操作系统并不能非常好地满足 CAN、LIN 总线的要求。例如，ECOS 虽然实时性很强，但对资源要求很高，需要很大的存储空间； $\mu$ CLinux 和  $\mu$ C/OS 则实时性和尺寸都不太能够满足。更重要的是，目前，很少有操作系统能够为 CAN 和 LIN 总线量身定做一套适应它们的通信系统，因此，我们设计了这个基于 CAN 和 LIN 总线的嵌入式多任务操作系统——FDCX08。

## 3 设计与实现

本实时操作系统基于 Motorola M68HC08 单片机设计，使用该处理器的软中断模式作为操作系统的系统态，在其中进行任务切换、通信服务、PV 操作、中断控制等系统操作。

本操作系统支持多至 8 个任务同时运行，采用优先级任务调度方法，支持抢先调度；支持任务间通信，具有邮局通信和信号量控制功能；支持 CAN 和 LIN 通信，每个任务即可与 CAN 总线上各节点的任何指定任务进行通信，也可与 CAN 节点上扩展的 LIN 节点的任何任务进行通信；具有定时时钟功能，可为各个任务提供定时和超时计数服务；支持 M68HC08 的各种中断，为各个任务提供中断处理服务；用户接口简单，通过系统调用为任务服务。图 2 为该系统的体系结构。

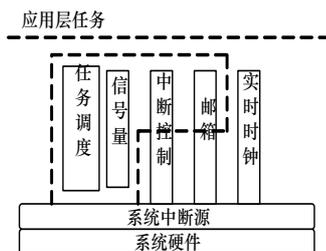


图 2 FDCX08 的体系结构

图中虚线部分在软中断中实现，其它部分在各自的中断服务子程序中实现。

下面从以下几个方面介绍该操作系统的设计和实现。

### (1)任务调度

FDCX08 最多支持 8 个任务，每个任务都有自己单独的任务现场，包括各个寄存器、堆栈指针、程序指针和堆栈区等，这样就可以使一个 CPU 由多个任务共享。

FDCX08 采取单队列优先级调度，每个任务一个优先级(0-7)，任务号大的优先级高，允许抢先操作。它保证该操作系统的硬实时性。

由于 CAN 和 LIN 总线对系统的实时性要求很高，因此，FDCX08 必须是一个硬实时系统。而对于 CAN、LIN 总线的

节点一般主要执行通信、数据采集和控制的任務，所以，系统支持 8 个任务就可以完全满足整个系统的应用。

### (2)任务通信控制

FDCX08 的通信控制支持以下 2 种方式：邮局通信和信号量。

1)邮局通信。邮局通信是一个强大的支持 CAN、LIN 总线及时通信的通信模块。支持操作系统之间的任务通过 CAN 和 LIN 总线进行通信。

一个任务需通信时，向系统申请一个信封，写入内容，发给其它任务(或其它节点的任务)。其它任务收到信封后，如不需回答，可把信封退还给系统，如要求回答，可用该信封写入内容，发还给原发信任务。

每一个信封中都含有 CAN 和 LIN 的节点信息及标明目的节点类型的标志位，使得该信封可以适应 CAN、LIN 节点之间的各种点对点通信方式(CAN 到 CAN, LIN 到 LIN, CAN 到 LIN 等)。而且由于在信封中设置了发送任务和接收任务的参数，因此可以实现从一个操作系统的 A 任务发送到另一个操作系统的 B 任务上，增加了系统的灵活性。

2)信号量。信号量是实现任务间通信的最常用的方法之一。一个信号量是一个整数变量 S，除了设置初值外，它只能通过 P 操作和 V 操作来改变。对于每一个信号量 S，都有一个对应的等待队列 Q。对于等待队列 Q，也有一个调度算法，它决定 V 操作时应该唤醒等待队列 Q 中的哪个任务。FDCX08 采用基于优先级的调度方法。

### (3)实时时钟

FDCX08 的实时时钟由定时器输出比较功能来实现一个单位时间 Tick，一般为 10ms。累计时间的功能在定时器中断中完成，时刻记忆实际的时间值。对于每个任务，都有一个定时时间表，定时初值表和超时计数器，以实现任务的定时、等待超时等功能。

这些保证了任务的硬实时性，使得每一个任务都可以周期性地执行，并且可以在 deadline 之前结束任务。这些都保证了 CAN、LIN 总线对实时性的要求。

### (4)中断处理

FDCX08 允许为任务设置与之对应的中断，每个中断最多只能对应一个任务。一个任务可对应多个中断。以 MC68HC08AZ60 为例，除了系统控制 6 个中断(软件中断、系统时钟中断、CAN、LIN 发送中断、CAN、LIN 接收中断等)以外，其它中断都由中断处理部分控制。

这些既保证了中断响应的及时性，又可以让中断在操作系统的控制之下。从而保证整个 CAN、LIN 总线的通信可以及时完成。

### (5)系统态的设计与实现

系统态的设计是 FDCX08 的核心，它利用处理器的软件中断(SWI)，实现了这个操作系统的核心。在系统的核心态中，操作系统将完成所有操作系统的核心任务，如任务切换、通信服务、PV 操作、中断控制等。由于软件中断是除了复位中断外最高优先级的中断，因此在系统运行过程中，任何程序都无法打断它，就可以将所有的核心任务放在软件中断中，这样，整个系统就被分为系统态和用户态，从而保证了整个操作系统的实现。系统运行过程中，遇到任何一个与系统相关操作或者事件(如信号量、等信、等中断、等定时、任务切换、有信、有信号量、有中断、定时到等)时，都需要进入系统态。整个系统态的软件流程见图 3。

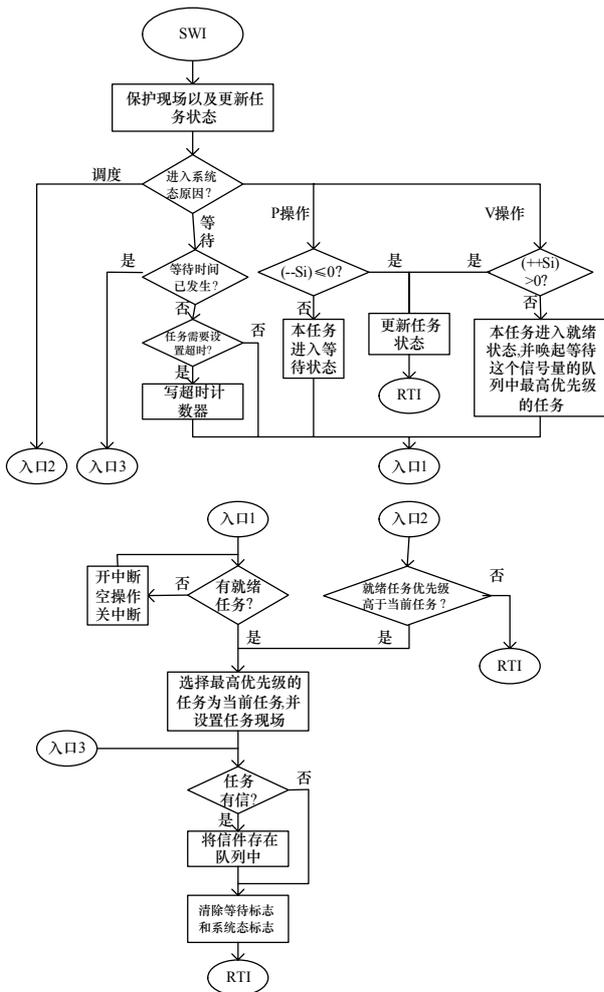


图3 系统态软件流程图

进入和退出系统态时，由寄存器 A 和 H: X 作为出入口参数，其中，A 代表进入系统态的原因，H: X 则代表信封或者信号量的指针(如果需要的话)。

#### 4 应用

目前，FDCX08 应用于一个通用控制系统中。该系统的主要目的是实现一个通用的控制系统，可以控制包括温度、流量、开关量等各种通用的对象，对于每种控制，都可以从多种 PID 算法中选择，并可以实现数据采集、报警、变送输出等通用功能。

##### 4.1 总体结构

如图 4 所示，该系统由一个上位的操作机(用现代的 HMS30C720 实现)和多个控制子系统(用 MC68HC908AZ60 实现)组成，整个网络由 CAN 总线连接。

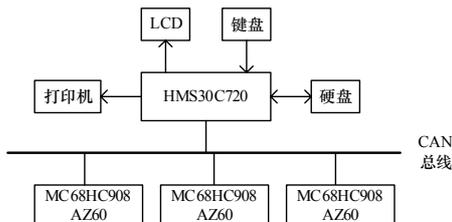


图4 通用控制系统的总体结构

##### 4.2 软件结构

上位机使用 Linux 作为操作系统，实现人机界面以及数据查询打印等功能。下位机则使用本文介绍的 FDCX08 作为

操作系统，整个软件应用了操作系统的各个方面，信号量、邮箱、中断控制、实时时钟等，充分发挥了 FDCX08 的优势。软件的结构如图 5 所示。

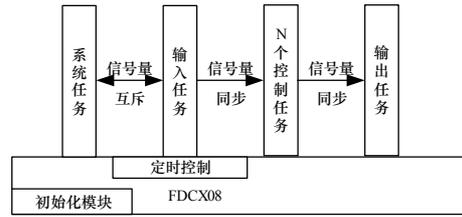


图5 控制子系统的软件结构

系统任务为最低优先级，主要完成 CAN 通信、报警以及系统检验任务；输入任务根据各个输入的采样周期，定时地采样输入值并送给指定控制任务；每个控制任务代表一个控制通道，根据输入值和控制算法得到输出值，并送给输出任务；输出任务根据设定将控制任务算出的输出值，通过特定的输出方式输出。

输入任务定时运行，并通过信号量使得在到达采样时间时，输入任务、控制任务和输入任务连续运行。由于设置参数和进行控制不能同时进行，因此用信号量互斥使得系统任务和输入任务不能同时进行。

系统启动以后，根据用户通过上位机的设定，系统定时进行 A/D 采样和读取开关量，如果需要得到虚拟输入(对于需要测量多个传感器参数经过计算才能得到实际值的物理量，比如流量，需要单独处理，因此把这种输入称为虚拟输入)，则将需要的采样值送入虚拟输入任务，然后将各个控制任务需要的输入值送给该任务，经过用户设定的控制算法得到输出值，按照用户设定的输出方式输出，并将各种数据以及结果送给上位机。如果出现异常情况，则报警。

#### 5 结论

FDCX08 在应用中表现出了很多特点，下面通过和目前流行的嵌入式操作系统  $\mu\text{C}/\text{OS-II}$  在以下方面的比较来说明。

(1)体积。在实现标准配置的情况下，FDCX08 只需要 2KB，而  $\mu\text{C}/\text{OS-II}$  一般需要 6KB~10KB。如果做最精简的剪裁， $\mu\text{C}/\text{OS-II}$  也需要 2KB，而 FDCX08 只需要 1KB。这样，对于工业控制或者汽车电子等对资源要求非常苛刻的领域控制来说，这种操作系统的优势是明显的。

(2)核心操作。表 1 中列出了 FDCX08 和  $\mu\text{C}/\text{OS-II}$  在操作系统核心操作方法使用上的异同。在任务调度机制上，FDCX08 为每个任务都设置了定时计时器和超时计时器，使得用户更加方便地实现应用程序的实时性。在通信机制上，FDCX08 则创造性地设计了一个可支持操作系统间通信的邮箱系统，使得操作系统更加接近于一个分布式操作系统，从而更加方便地应用于工业控制或者汽车电子等控制网络中。

表 1  $\mu\text{C}/\text{OS-II}$  与 FDCX08 在核心操作方法使用上的比较

	任务调度机制		同步与通信	
$\mu\text{C}/\text{OS-II}$	固定优先级占先	不支持时间片轮转	邮箱，信号量	消息队列
FDCX08		每个任务都没有定时超时计时器		支持操作系统之间的通信

(3)任务切换时间和中断延迟时间。任务切换时间和中断延迟时间是评估 RTOS 性能的两个重要指标。所谓任务切换时间，即系统发生任务切换时，保存和恢复上下文的时间，它可以反映出 (下转第 279 页)

参与是搞好信息资源规划的关键因素。

搞好装备保障的信息资源规划，从源头抓好数据标准，建立装备保障信息系统的功能模型和数据模型，用以控制、指导和协调各项信息工程项目成功的实施，就是最关键的战略。这一战略的制定与实现，是真正的“一把手工程”<sup>[5]</sup>。强调装备保障信息化建设是“一把手工程”，就是强调装备保障部门高层领导要重视信息化工作和参与信息化工作。

有经验证明，装备保障部门高层领导的倡导、支持和亲自参与，是信息资源规划成功的保证。装备保障部门高层领导只有关心并亲自参与信息资源规划，才能理解和把握信息资源规划方案，才能对装备保障的信息化建设有指挥权。

#### 4.2 需求分析与系统建模紧密结合起来

信息化建设中一个长期没有解决好的问题，就是用户讲不清需求，分析人员所做的需求分析并不能真正反映用户的需求<sup>[6]</sup>。需求分析是系统建模的准备，信息资源规划必须要强调将需求分析与系统建模紧密结合起来，系统建模是用户需求的定型和规范化表达。在进行装备保障信息资源规划的时候，要经过详细的调研分析，调研分析职能域之间、职能域内部、职能域与外单位间的数据流，而且要做出流量的估算。只有这样才能进行科学的综合分析，得出模型化的认识，使装备保障部门领导、业务人员和技术人员在信息化建设“要做什么”的问题上达成共识。

#### 4.3 做好基础的数据标准化工作

信息资源规划的基本核心是信息资源管理（IRM）。信息资源管理的目标是提高管理的效益，即追求高效、实效和经济<sup>[7]</sup>。装备保障只有从战略高度开发信息资源、科学合理地管理信息资源、充分有效地利用信息资源，才能在竞争中取胜。而信息资源管理首先需要明确的是数据管理，没有卓有成效的数据管理，就没有成功高效的数据处理，更建立不起来全装备保障的计算机信息系统。而数据管理工作必须从最小的信息单元——数据元素的标准化做起<sup>[8]</sup>。

数据标准化管理是目前装备保障信息化建设中存在的一个主要问题，表现在数据定义和编码没有统一标准，对数据

库和数据的管理也基本上是各自独立运作，没有统一的规程和要求。不同的程序员、或者同一个程序员在不同的时期，对同一个数据对象起了很多不同的名称。

面对装备保障应用系统复杂、用户多、数据量庞大这一特点，需要设立数据管理部门来集中控制和统一管理数据的定义和结构的规范化，指导协调信息系统开发和使用人员执行标准、规范和行之有效的办法，就是在信息资源规划的过程中，结合数据流的调研分析，做好基础的数据标准化工作，进而落实到数据库的结构标准化。

### 5 结束语

信息化建设是装备保障发展的重要战略之一，而信息化的最大效益来自信息最广泛的共享、最快捷的流通和对信息深层次的挖掘。只有通过装备保障 IRP、梳理业务流程、理顺信息需求，建立信息资源管理基础标准和信息系统模型，用这些标准和模型来衡量现有的各种应用系统，才能实现装备保障信息化建设的跨越式发展。

#### 参考文献

- 1 任 凯. 医院信息资源的孤岛现象及对策[J]. 医院信息, 2003, 16 (6).
- 2 刘世利. 高校管理信息化建设中的信息资源规划[J]. 辽宁师范大学学报, 2004, 27(1).
- 3 高复先. 信息资源规划——信息化建设基础工程[M]. 北京: 清华大学出版社, 2002.
- 4 高复先. 数据管理——信息系统成功建设的基础[Z]. 沈阳: 东北电力试验研究院, 1992.
- 5 企业信息资源规划的几点观点[Z]. <http://www.MISCHINA.COM/99CLUB/BBS>.
- 6 Martin M. 信息工程与总体数据规划[M]. 高复先, 吴曙光, 译. 北京: 人民交通出版社, 1989.
- 7 朱翠屏, 马晓东, 蔡文海. 城市公安信息资源规划[J]. 中国人民公安大学学报, 2004, (4).
- 8 肖 明. 信息资源管理[M]. 北京: 电子工业出版社, 2002.

(上接第 267 页)

RTOS 执行任务的速度; 而中断延迟时间, 则是中断发生起, 到执行中断处理程序的第 1 条指令所用的时间, 它可以反映出 RTOS 对外界变化的反应速度。

表 2  $\mu\text{C}/\text{OS}$  与 FDCX08 的相对任务切换时间和相对中断延迟时间

	相对任务切换时间	相对中断延迟时间
$\mu\text{C}/\text{OS-II}$	8.5 $\mu\text{s}$	2.5 $\mu\text{s}$
FDCX08	7.5 $\mu\text{s}$	1 $\mu\text{s}$

为了便于比较, 将  $\mu\text{C}/\text{OS-II}$  也移植到 M68HC08 上, 并配置相同的总线频率(8MHz)。由于  $\mu\text{C}/\text{OS-II}$  也使用了软件中断作为系统核心态, 因此二者进入核心态的响应时间相同; 同理, 其它中断的响应时间也相同。因此, 可以用相对时间来计算: 对于任务切换时间, 可以用在软中断中任务调度程序的时间来计算; 而中断延迟时间, 可以用进入中断子程序之后执行用户中断服务程序之前需要执行的系统程序的执行时间来计算。表 2 为这 2 种操作系统相对任务切换时间和相

对中断延迟时间的比较。

通过比较发现, FDCX08 具有体积小、实时性强、功能更加强大(更适用于控制网络)等特点。而工业控制、汽车电子等利用 CAN、LIN 总线控制领域对资源要求非常苛刻, 而且往往是一种网络式的协同工作, 所以, FDCX08 更加适合于这种使用 CAN 和 LIN 总线的应用场合。

#### 参考文献

- 1 Wolf W. Computers as Components: Principles of Embedded Computing System Design[M]. Morgan Kaufman Publishers, 2001.
- 2 张友得, 涂时亮, 陈章龙. M68HC08 系列单片机原理与应用[M]. 上海: 复旦大学出版社, 2001.
- 3 郭宽明. CAN 总线原理和应用系统设计[M]. 北京: 北京航空航天大学出版社, 1996.
- 4 Motorola Inc.. Local Interconnect Network Demonstration[Z]. 2000.
- 5 Labrosse J J.  $\mu\text{C}/\text{OS-II}$  源码公开的实时嵌入式操作系统[M]. 邵贝贝, 译. 北京: 中国电力出版社, 2001.
- 6 中国软件评测中心. 漫谈嵌入式实时操作系统与测试[R]. 2002.