

基于 TMS320F2812 的

CAN 总线通信研究

朱清祥 聂汉平 长江大学电信学院 434023

Research on CAN bus Communication Based on TMS320F2812
Zhu Qingxiang, Nie Hanping
College of Electronics and Information, Yangtze University, Jingzhou, Hubei, China

摘要

CAN 总线是一种串行通讯协议,适用于分布式控制系统的底层网络通信。由于标准的 CAN 协议为同一节点的信息分配固定的优先级,网络控制的实时性得不到保证。本文提出了一种利用 TMS320F2812 的增强型 CAN 模块实现网络监控的方法,通过标志符为 CAN 总线提供不同的优先级,在确保网络控制实时性的同时兼顾了 CAN 总线利用率,取得了较好的效果。在波特率高达 1Mb/s 的条件下,数据传输稳定、可靠,延迟时间少。

关键词

DSP; CAN 总线; 实时性

中图分类号: TP336

文献标志码: A

Abstract

CAN bus is a serial communication protocol and adapts to bottom layer communication of distributed control system. It can't get the good performance in network control system because all the information of a node has the same priority. The article puts forward a method by which the network control is realized using the enhanced CAN module of TMS320F2812, which provide different priority for CAN bus with identification. The real-time performance is ensured at the high utilization ratio of CAN bus and it get good performance. In the case of 1Mb/s (Baud Rate), data transmission is stable and reliable with little delay.

Key words

DSP; CAN bus; real-time performance

1 引言

CAN (Controller Area Network) 总线是一种有效支持分布式控制或实时控制的串行通信网络,在 20 世纪 80 年代由德国 Bosch 公司提出并用于汽车内部测试

和控制仪器之间的数据通信。而后,世界上许多著名的汽车制造商都开始采用 CAN 总线来实现汽车内部控制系统的数据通信。由于 CAN 本身的特点,其应用范围目前已不再局限于汽车行业,而面向过程工业、机械工业、纺织工业、农用机械、机器人、数控机床、医疗器械等领域发展。现在, CAN 总线已形成 ISO-11898 CAN 高速应用国际标准,并已成为几种最有前途的现场总线之一。

TMS320F28x 系列是目前控制领域性能最高的 32 位定点 DSP 芯片,具有精度高、速度快、集成度高等特点,为控制领域提供了高性能解决方案。它集成了增强型 CAN 总线通信接口、16 路 PWM 输出、16 路 12 位 ADC 转换器、可编程死区控制功能、可编程空间 PWM 控制方式等电机控制所特需的资源。TMS320F2812 数字信号处理器集成的增强型 CAN 总线通信接口与 CAN2.0B 协议完全兼容。这种 CAN 控制器集成了 CAN 总线的物理层和数据链路层,控制器的链路层能自动完成 CAN 可靠的错误检测以及处理。它的 CAN 模块有 32 个邮箱,提供了数字回路自测模式。

本文提出了应用 TMS320F2812 数字信号处理器作为控制器芯片,利用它的 eCAN 模块实现 CAN 总线通信。

2 硬件设计

TMS320F2812 上的 eCAN 模块集成了 CAN 控制器,提供了完整的 CAN 协议,减少了 CPU 的开销,仅需外置收发装置。这里采用 PHILIP 公司的 PCA82C250 作为收发器,

PCA82C250 是 CAN 协议控制器和物理总线的接口,此器件对 CAN 总线提供差分发送能力,对 CAN 控制器提供差分接收能力。由于收发器采用 5V 供电,在收发器和 DSP 之间加高速光耦 6N137 以实现电平转换,并起到使 DSP 和 CAN 总线隔离,达到保护 DSP 芯片的目的。电路原理图如图 1 所示。

3 CAN 总线通信的软件设计

各种监控数据通过 CAN 总线进行定时发送和中断接收。定时发送通过 DSP 芯片的定时器 0 实现,定时的时钟周期可以调节。中断接收的响应速度快,避免了丢包现象。软件设计主要分成初始化,定时器中断服务程序和 CAN 总线中断接收服务程序三个部分。

3.1 初始化

DSP 初始化的流程图如图 2 所示。它包括系统初始化, PIE 模块初始化,定时器初始化及 eCAN 模块初始化等几个部分。系统初始化的主要目的是初始化系统控制寄存器、看门狗、时钟,使 DSP 运行在已知的状态。锁相环 (PLL) 模块主要是用来控制 DSP 内核的工作频率,外部提供一个参考时钟输入,经过锁相

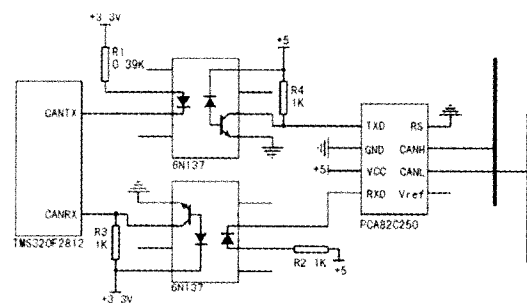


图 1 TMS320F2812 的 CAN 接口电路原理图

环(PLL)倍频或分频后提供给DSP内核。根据需要频率给锁相环控制寄存器(PLLCCR)的DIV位写入相应的值就可实现。这里设置DIV位为10,输入时钟为30MHz,提供给DSP内核的工作频率为150MHz。

外设中断扩展模块(PIE)中多个中断源复用几个中断输入信号,PIE最多可支持96个中断,以8个中断为一组,共有12组中断。外设中断扩展模块(PIE)的初始化由禁止和清除所有中断、初始化PIE控制寄存器和初始化中断向量表三部分组成。配置中断模块时先初始化PIE控制寄存器和PIE中断向量表。系统中PIE向量表存放着每个中断服务程序的地址,每个中断都有自己的中断向量,初始化中断向量表是要定位中断向量。然后设置中断服务的入口地址,根据中断向量表中所定义的地址,把相应的中断向量指向所要执行的子程序。

TMS320F2812中有3个32位CPU定时器,其中定时器1和定时器2是预留给实时操作系统使用的,用户在应用程序中可使用定时器0。这里的定时器模块初始化仅需要对定时器0进行初始化,即设置定时器周期寄存器和定时器预定标寄存器的值。

CAN模块在使用之前,必须进行初始化,并且只有在CAN模式下才能初始化。初始化eCAN模块包括:配置引脚、清除标志位、配置位时序参数和配置邮箱。CAN模块初始化流程图如图3所示。

CANTX和CANRX引脚可以作为CAN的通信接口引脚,也可以作为通用IO使用,是否作为通用IO使用由功能选择控制寄存器(GPFMUX)的位6和位7控制。位时序配置寄存器用来配置节点的网络时序参数,在使用CAN模块之前必须配置该寄存器,且只有初始化模式时才能改变该寄存器。复位后主控制寄存器(CANMC)的改变配置请求位(CCR)为0,eCAN处于正常工作方式;错误状态寄存器(CANES)的改变配置使能位(CCE)也为0,这时CPU不能对控制寄存器进行写操作。在CCR位置1后,eCAN处于要求配置的状态,等到CCE位置1后CPU才可以对定时器配置寄存器(CANBTC)进行配置。对CANBTC寄存器的配置,主要是设置通信波特率预设值(BRPreG)、时间段1(TSEG1)和时间段2(TSEG2)的值来确定CAN通信波特率。

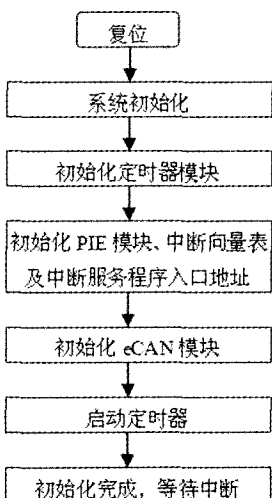


图2 初始化流程图

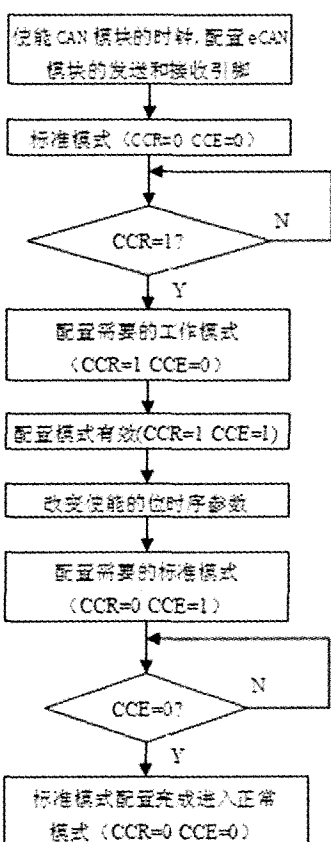


图3 eCAN模块初始化流程图

状态,等到CCE位置1后CPU才可以对定时器配置寄存器(CANBTC)进行配置。对CANBTC寄存器的配置,主要是设置通信波特率预设值(BRPreG)、时间段1(TSEG1)和时间段2(TSEG2)的值来确定CAN通信波特率。

3.2 定时发送数据

由于电动机监控的需要,电动机控制器被要求能够不断发送转速等反应电动机运行状态的数据,故应用定时器中断功能实现数据的定时发送。定时器中断服务子程序主要完成采集将要发送的数据及通过CAN总线发送数据给监控计算机或控制器。定时器中断服务子程序流程图如图4所示。定时器的定时周期由32位的定时器周期寄存器(PRD)和16位的定时器预定标寄存器(TDDR)共同决定,PRD乘以(TDDR+1)个定时器时钟源周期即为定时器周期。启动发送命令即向相应发送请求置为位寄存器写1。向相应的响应寄存器(PIEACK1)写1即可清除定时器0的中断。

3.3 中断方式接收数据

CAN总线中断接收服务子程序完成数据的接收,并根据接收的数据做出相应的处理。接收的数据即表示电机的启动、停止、加速、减速、调整加速度等控制指令。向相应邮箱的接收消息挂器寄存器写1即可清除引起中断的标志位。CAN总线中断接收服务子程序流程图如图5所示。当CAN模块检测到有需要接收的数据时,产生中断信号向CPU申请中断这就保证了接收数据的实时性,为保证电动机的实时控制提供了条件。将两片DSP分别设置成主控制器、从控制器,主控制器定时发送数据,从控制器中断接收数据。在接收到数据后立即发送数据,以表示已经接收到数据,这样可以得出从主控制器发送一帧8字节数据到从控制器的时间。经多次实验,在波特率为1Mb/s时这一时间不超过270μs,即发送一帧数据的时间不超过135μs,这符合实时控制的要求。

4 CAN总线通信的实时性

TMS320F2812的eCAN模块有多达32个邮箱,每个邮箱均可以被配置为发送或接收邮箱,这使每个节点的信息传输更加灵活、方便,但当CAN总线上节点较多时,就容易造成网络过载或总线拥塞。为了确保重要业务数据包不延迟或丢失,需要在原有的网络带宽基础上为重要业务信息提供相对稳定的传输带宽,同时保证网络的高效运行。本设计通过分配给一个节点多个报文标志符来实现数据的不同优先级,从而能够及时发送高优先级的数

下转第123页

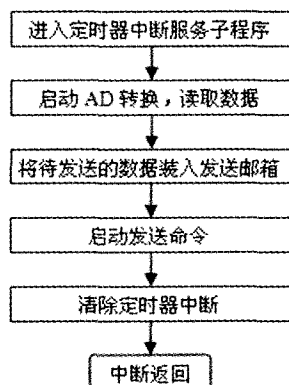


图4 定时器中断服务子程序流程图

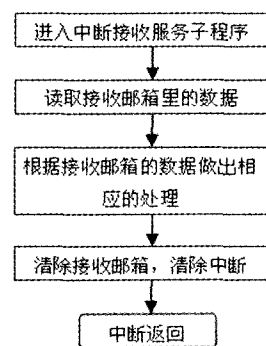


图5 CAN总线中断接收服务子程序流程图

是忘记那些变量是不是复数。

3.2、偶然地使用了保留字

一些字不能作为变量名,因为它们已经被 JavaScript 使用。例如,不能定义一个叫“if”的变量,因为它实际上是 JavaScript 的一部分——如果使用“if”,你会遇到各种麻烦。当你因为使用命名为“if”的变量而变得疯狂时,一个叫做“document”的变量是很诱人的。不幸的是,“document”是一个 JavaScript 对象。另一个经常遇到的问题是把变量命名为“name”(窗体元素有“names”属性)。把变量命名为“name”不会总出问题,只是有时——这会更使人迷惑——这就是避免使用“name”变量的原因。不幸的是,不同的浏览器有不同的保留字,所以没有办法知道该回避哪些字。最安全的办法是避免使用已经成为 JavaScript 一部分的字和 HTML 使用的字。如果你因为变量遇到问题,并且不能发现哪儿错了,试着把变量改个名字。如果成功了,你或许就避开了保留字。

3.3、记住在逻辑判断时应该用两个等号

一些浏览器能捕获这种错误,有些却不能。这是一种非常常见的错误,但是如果浏览器不能替你指出来,你就很难发现。下面是一个这种错误的例子:

```
var the_name = prompt("what's your name?", "");
if (the_name = "the monkey")
{
    alert("hello monkey!");
} else {
    alert("hello stranger.");
}
```

这段代码将产生“hello monkey!”警告对话框——不管你在提示里敲的是什么——这不是我们希望的。原因是在 if-then 语句中只有一个等号,这句话告诉 JavaScript 你想让一件事等于另一件。假设在提示中敲的是“robbie the robot”。最开始,变量 the_name 的值是“robbie the robot”,但是随后 if 语句告诉 JavaScript 想把 the_name 设为“the monkey.”。于是 JavaScript 很高兴地执行命令,送一个“true”消息给 if-then 语句,结果警告对话框每次都出现“hello monkey!”。这种阴险的错误会使你发疯,所以注意使用两个等号。

3.4、偶然给变量加上了引号,或忘了给字符串加引号

JavaScript 区分变量和字符串的唯一

方法是:字符串有引号,变量没有。下面有一个明显的错误:

```
var the_name = 'koko the gorilla';
alert("the_name is very happy");
```

虽然 the_name 是一个变量,但是程序还会产生一个提示“the_name is very happy,”的警告对话框。这是因为一旦 JavaScript 看见引号包围着某些东西就不再考虑它,所以当你把 the_name 放在引号里,你就阻止了 JavaScript 从内存中查找它。

下面是一个不太明显的此类错误的扩展:

```
function wakeMeIn3()
{
    var the_message = "Wake up! Hey! Hey! WAKE UP!!!!";
    setTimeout("alert(the_message);", 3000);
}
```

这里的问题告诉 JavaScript 三秒后执行 alert(the_message)。但是,三秒后 the_message 将不再存在,因为已经退出了函数。这个问题可以这样解决:

```
function wakeMeIn3()
{
    var the_message = "Wake up!";
    setTimeout("alert(' ' + the_message + ' ');", 3000);
}
```

把 the_message 放在引号外面,命令“alert('Wake up!');”由 setTimeout 预定好,就可以得到你想要的。

这只是一些可能在你的代码中作祟的很难调试的错误。一旦发现了它们,就有不同的或好或差的方法来改正错误。

4、结束语

以上只是我们在 JavaScript 编程中遇到的最常见的错误,其实在 JavaScript 中我们也经常碰到在其他脚本语言编程中遇到的错误,不过由于和其他语言基本上没什么大的区别,所以本文就不在详细论述。有什么不妥之处,还请各位同行斧正。

参考文献

- [1]. 王炜. Web 编程丛书——JavaScript 编程指南. 电子工业出版社
- [2]. Nicholas C. Zakas. JavaScript 高级程序设计. 人民邮电出版社
- [3]. <http://www.yfdmt.com>

作者简介

陈华 1978-04-02 女 汉族 山东省济宁市助教 学士 网络技术

上接第 121 页

据。监控控制器的最高优先级保证了其对总线的控制;至于 CAN 总线上节点的其他控制信息,则为非周期性数据的优先级高于周期性数据的优先级。

邮箱的超时功能可以确保重要数据的及时发送。在网络监控系统中,当总线上的数据总量超过了它的承载能力时,则需要抛弃部分相对不重要的数据,邮箱的超时功能满足了这种要求。为了确定所有的信息都在预定的时间里成功发送或接收,每个邮箱都有它自己的超时寄存器(MOTO)实现超时控制功能。因此当 CAN 总线上待发送的数据总量超过它的传输能力时,控制器能够自动抛弃部分周期性数据从而保证总线的正常通信。

5 结论

和普通 CAN 控制器相比,TMS320F2812 的 eCAN 模块具有更多的邮箱,有超时功能等增强型功能,在进行 CAN 总线通信时,数据传输更加灵活方便,数据量更大,可靠性更高,功能更加完备。实践表明,基于 eCAN 总线的多节点网络实时监控性好,能分别处理同一节点不同优先级的数据,在数据通信速率高达 1Mb/s 的情况下实现了数据的可靠传输,延迟时间小,因此,TMS320F2812 增强型 CAN 总线满足多节点实时控制的要求,在分布式控制系统中有广阔的应用前景。

参考文献

- [1] TMS320x281x, 280x Enhanced Controller Area Network (eCAN) Reference Guide (Rev. D). Texas Instruments. 08 Nov 2005
- [2] 邓竹莎等. 基于 CAN 实时应用的可靠调度性研究. 成都: 计算机应用. 2006(3)
- [3] 史久跟等. CAN 现场总线系统设计技术. 北京: 国防工业出版社. 2004
- [4] 苏奎峰等. TMS320F2812 原理与开发[M]. 北京: 电子工业出版社. 2005
- [5] 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

作者简介

朱清祥 (1965-) 男, 讲师, 硕士。1985 年毕业于华中工学院自动化专业, 获学士学位。1997 年获华中理工大学通信与电子系统专业硕士学位。