

文章编号:1671-251X(2009)06-0052-04

CAN总线波特率设置方法及误差分析

于月森, 钱振华, 彭利明, 刘剑

(中国矿业大学信电学院, 江苏徐州 221008)

摘要: CAN总线是一种有效支持分布式控制和实时控制的、多主的异步串行通信网络。文章以C8051F060单片机内集成的CAN控制器为例,详细介绍了波特率设置的方法,分析了波特率设置的误差来源,并给出了波特率误差的消除方法。

关键词: CAN总线; 波特率; 误差分析; 单片机; C8051F060

中图分类号: TP336 **文献标识码:** A

Setting Method of Baud Rate of CAN Bus and Its Error Analysis

YU Yue-sen, QIAN Zhen-hua, PENG Li-ming, LIU Jian

(School of Information and Electrical Engineering of CUMT., Xuzhou 221008, China)

Abstract : CAN bus is a kind of network with asynchronous and serial communication which can

收稿日期: 2009-02-12

作者简介: 于月森(1978-),男,博士,讲师,现主要从事工业自动化方面的教学与研究工作。Tel: 0516-83995808; E-mail: cumtyys@126.com

的 Project/make 对工作区的项目进行编译链接,没有错误即可生成可执行代码。将 JTAG 调试电缆的一端与 PC 机的并口相连,另一端与电路板上的 JTAG 相连。先启动 H-JTAG 调试代理软件,当其检测到目标 CPU 后,再选择工具栏中的 Project/debug 将程序通过 JTAG 装入外扩的 SRAM 中运行。

通过 IAR EWARM 自带的 IAR KA 插件可以在调试时看到如图 6 所示的信息界面,且此时连接在 AT91M40800 的 P17 口上的小灯被循环点亮和熄灭,这说明 AppStart Task() 任务成功建立且在 μC/OS-II 内核调度下得以成功运行。

*	Name	Ref	Prio	State	Dly	CtxSw	Stk Ptr	Max%Cur%	Max Cur	Size	Starts @	Ends @
Start Task	2	5	Dly	8	39	100201A0	19%	19%	100	100	512	10020204 10020004
μC/OS-II Stat	1	62	Dly	6	38	1002069C	18%	18%	96	96	512	100206FC 1002004C
μC/OS-II Idle	0	63	Ready	0	77	100208C0	14%	14%	72	72	512	10020908 10020708

图 6 调试时 IAR KA 插件所显示的信息界面

4 结语

应用层的测试程序通过调用相应的 μC/OS-II 内核函数成功建立了循环点亮和熄灭小灯的任务 AppStart Task(),且由图 6 中 CtxSw 列可得出:当

AppStart Task() 任务调用 μC/OS-II 内核函数 OSTimeDly() 时, μC/OS-II 内核成功进行了任务切换;对所建立的 3 个任务的堆栈空间分配合适,说明移植 μC/OS-II 成功, μC/OS-II 内核进行了多任务调度操作。

μC/OS-II 在 AT91M40800 上的移植成功,为多总线通信协议转换器构建了稳定可靠的开发平台,缩短了开发周期,具有一定的实用价值。

参考文献:

- [1] 史丽萍,王攀攀,张广龙,等.多协议以太网网关在煤矿综合自动化系统的研究与开发[J].继电器,2007,35(24):43~47.
- [2] 朱义军,杨育宏,赵凯,等.A T91 系列 ARM 核微控制器体系结构与开发实例[M].北京:北京航空航天大学出版社,2005.
- [3] LABROSSE J J. 嵌入式实时操作系统 μC/OS-II[M].2 版.邵贝贝,译.北京:北京航空航天大学出版社,2003.
- [4] 徐爱军.IAR EWARM 嵌入式系统编程与实践[M].北京:北京航空航天大学出版社,2006.

effectively support distributed control and real-time control and multi-master. Taking integrated CAN controller in C8051F060 single-chip microcomputer as example, the paper introduced the setting method of baud rate in details. It analyzed error sources of baud rate setting and gave several ways for eliminating the error of baud rate.

Key words: CAN bus, baud rate, error analysis, single-chip microcomputer, C8051F060

0 引言

CAN (Controller Area Network) 总线即控制器局域网, 是一种有效支持分布式控制和实时控制的、多主的异步串行通信网络, 具有高性能、高可靠性, 越来越受到人们的重视。CAN 总线规范已经被国际标准化组织制订为国际标准 ISO11898, 并得到众多半导体器件厂商的支持, 他们推出了各种集成有 CAN 协议的产品^[1]。C8051F060 是美国 CYGNAL 公司生产的内部集成有 BOSCH CAN 控制器的混合信号系统级芯片。CAN 总线的数据传输速率最高可达 1 Mbps, 每个 CAN 节点的时钟信号都由自己的晶振产生, 可独立进行位定时参数的设置, 这样即使网络中节点之间的时钟周期不一样时仍可获得相同的位速率。这些晶振的频率不是完全固定的, 受温度和电压的影响。但是只要这些变化在一定的晶振频率容错范围内, 可通过比特位的同步补偿机制进行补偿, 从而进行正常的通信; 如果这些变化超过一定的晶振频率容错范围, 则通信失败。本文以 C8051F060 CAN 控制器为例, 详细说明波特率的设置方法及其误差分析。

1 位周期组成

系统时钟 f_{sys} 来自 C8051F 兼容的 CIP251 内核的系统时钟 (SYNCK), 由于 CAN 通信的高精度要求, 一般需要使用外部振荡器(如石英晶体)。

波特率 F_{bit} 是指单位时间内所传输的数据位的数量, 一般取单位时间为 1 s。波特率由通信线上传输的 1 个数据位周期的长度 T_{bit} 决定, 如式 (1) 所示:

$$F_{bit} = 1 / T_{bit} \quad (1)$$

时间量子 t_q 是 CAN 控制器能识别的最短时间单元。位时序参数通常用时间量子 t_q 的整数倍给出。在 C8051F 系列内核中, 时间量子 t_q 不仅与系统时钟有关, 还与波特率预分频比例因子 BRP 有关, 其计算公式如式(2)所示:

$$t_q = (BRP + 1) / f_{sys} \quad (2)$$

C8051F060 CAN 控制器的一个标称位时间由

4 个部分组成: 同步段 (*Sync_seg*)、传播时间段 (*Prop_seg*)、相位缓冲段 1 (*Phase_seg1*)、相位缓冲段 2 (*Phase_seg2*), 如图 1 所示。

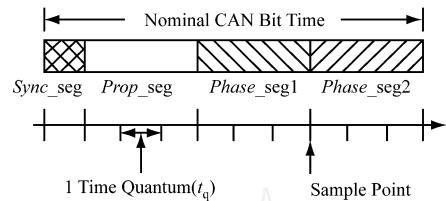


图 1 标称位时间组成图

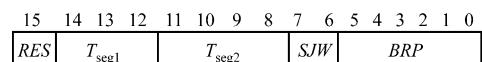
每个时间段由数目在一定范围内的可编程的时间量子 t_q 组成, 如表 1 所示。同步段用于补偿总线上不同节点的时序, 同步段开始时产生一个由高电平到低电平的跳变沿。传播段用于补偿网络的物理延时时间; 相位缓冲段 1、相位缓冲段 2 用于补偿边沿阶段的误差, 这 2 个段可通过重新同步来加长或缩短。采样点 (Sample Point) 是读总线电平并解释各位值的一个时间点, 位于相位缓冲段 1 之后^[2]。

表 1 各时间段参数范围表

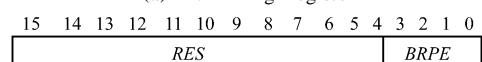
参数	范围	说明
<i>BRP</i>	[1..32]	定义时间量子长度
<i>Sync_seg</i>	1 t_q	固定长度
<i>Prop_seg</i>	[1..8] t_q	补偿物理延时
<i>Phase_seg1</i>	[1..8] t_q	在同步时可能被延长
<i>Phase_seg2</i>	[1..8] t_q	在同步时可能被缩短
<i>SJW</i>	[1..4] t_q	同步跳转宽度

2 C8051F060 CAN 控制器波特率设置

C8051F060 CAN 控制器的波特率设置主要是设置 2 个 16 位寄存器: Bit Timing Register 与 BRP Extension Register, 各位的定义如图 2 所示。



(a) Bit Timing Register



(b) BRP Extension Register

图 2 Bit Timing Register 与 BRP Extension Register 的定义示意图

对于 Bit Timing Register ,*RES* 为保留位 ; $T_{\text{seg1}} = \text{Phase_seg1} - 1$,4 位 , 可设置值为 $0x1 \sim 0xF$; $T_{\text{seg2}} = \text{Phase_seg2} - 1$,3 位 , 可设置值为 $0x0 \sim 0x7$; SJW ,2 位 , 可设置值为 $0x0 \sim 0x3$; BRP ,5 位 , 用于对 f_{sys} 预分频 , 可设置值为 $0x00 \sim 0x3F$ 。因为 T_{seg1} 、 T_{seg2} 、 SJW 与 BRP 都采用二进制数表示 , 所以要比实际使用值少 1。 BRP Extension Register 主要用在 f_{sys} 很高的情况 , 但是要求的 CAN 波特率又很小 , 如果 BRP 设置到 $0x3F$ 仍然不能满足要求 , 则用该寄存器扩展预分频的值。 $BRPE$ 为 4 位 , 作为分频数中的 MSB 部分 , BRP 为 6 位 , 作为分频数中的 LSB 部分 , 2 者合用共 10 位 , 最大可产生分频数为 1 023 , 即实际分频数最大可达 1 024。

通常根据需要的波特率设置波特率控制寄存器。首先根据目标波特率 F_{bit} 计算传播 1 bit 的时间 T_{bit} , 即标称位时间 ; 再根据系统的晶振频率和 BRP 的值计算时间量子 t_q ; 接着计算标称位时间 T_{bit} 内含的时间量子数 n (T_{bit} 除以 t_q 四舍五入得到)。一般 n 的取值范围在 4 ~ 25 之间 , 如果不在该范围内 , 可通过调整晶振频率和 BRP 的值来使之满足条件。标称位时间的第一段定义为同步段 (*Sync_seg*) , 它由固定的 $1 t_q$ 时间组成 ; 标称位时间的第二段定义为传播时间段 (*Prop_seg*) , 它的长度由系统的传播延时决定。系统的传播延时由信号在 CAN 总线的传输延时和 CAN 收发器内部延时 2 个部分组成。CAN 收发器内部延时一般取 150 ns , 也可查询具体的收发器参数获得真实值。信号在线路上的传输延时由线路的最大长度决定。根据计算的延时时间可得到传播时间段 (*Prop_seg*) 内包含的 t_q 数 , 线路传输延时的裕量由延时时间除以 t_q 向上取整得到。现在剩下的时间 (标称位时间 - 同步段 - 传播时间段) 分配给相位缓冲段 1 (*Phase_seg1*) 和相位缓冲段 2 (*Phase_seg2*) , 分配的原则 : 如果 $\text{Phase_seg1} + \text{Phase_seg2}$ 为偶数 , 则 $\text{Phase_seg1} = \text{Phase_seg2}$, 否则 $\text{Phase_seg2} = \text{Phase_seg1} + 1$ 。相位缓冲段 2 的最小长度不应小于控制器的信息处理时间 , 通常情况下 , 应该大于 $2 t_q$ 。同步跳转宽度 SJW 由 $SJW = \min(4, \text{Phase_seg1})$ 确定。该参数用于确定写入到位定时寄存器中的数值和确定所期望的振荡器误差。到此为止 , 所有参数已经确定完毕 , 最后就是写入寄存器。写入到寄存器中的值可用式(3)计算 :

$$(\text{Phase_seg2} - 1) \& (\text{Phase_seg1} + \text{Prop_seg} - 1) \\ \& (SJW - 1) \& (BRP - 1) \quad (3)$$

式中 : & 为将十进制数转换为二进制数后把二进制数连接起来的符号。

3 振荡器容错范围

在 CAN 总线通信中 , 通信双方应该采用相同的传输速率 , 即双方应该使用相同的传输波特率。然而在某些情况下通信双方不能正常、准确地接收和发送数据 , 这与波特率的设置和系统使用的振荡频率有着很重要的关系。波特率设置恰当与否不仅关系着通信的成败 , 而且对通信性能的好坏也起着决定性的作用。一般情况下 , 如果硬件连接无误 , 通信失败的主要原因在于总线定时寄存器设置不当。在实验中 , 还出现过因单片机的时钟电路中晶振严重漂移而导致的通信失败的情况 , 所以很有必要对振荡器的容错范围进行讨论。

当 CAN 总线协议从 1.1 版本过渡到 1.2 版本时 , 振荡器的容错范围已经扩大了 , 振荡器最大容差从 0.5 % 增加到了 1.58 %。当波特率超过 125 kbps 时不能使用陶瓷谐振器 , 只能使用石英晶振^[3]。笔者用 $f_{\text{CLK max/min}}$ 表示参考时钟频率满足通信要求的最大值或最小值 , 用 $f_{\text{CLK nom}}$ 表示参考时钟额定值 , 则容错范围 df 可以表示为

$$df = \left| \frac{f_{\text{CLK max/min}} - f_{\text{CLK nom}}}{f_{\text{CLK nom}}} \right| \quad (4)$$

容错范围的大小由相位缓冲段 1 (*Phase_seg1*) 、相位缓冲段 2 (*Phase_seg2*) 、同步跳转宽度 SJW 及标称位时间 T_{bit} 的比例决定。最大的容错范围由式(5)、式(6)决定 :

$$df \leq \frac{\max(\text{Phase_seg1}, \text{Phase_seg2})}{2 \times (13 \times T_{\text{bit}} - \text{Phase_seg2})} \quad (5)$$

$$df \leq \frac{SJW}{20 \times T_{\text{bit}}} \quad (6)$$

df 不仅必须同时满足式(5)、式(6) , 而且还必须满足 :

$$df \leq 1.58 \%$$

如果不满足以上条件 , 可通过调整晶振频率或波特率寄存器的方法来使之满足条件。

此外 , 在计算标称位时间 T_{bit} 内含有的时间量子数 n 时采用四舍五入的取整方法同样会产生误差。例如 : 要求的通信速率为 1 Mbps , 选取的系统时钟为外部 22.118 4 MHz 晶振经过 2 分频整形以后的时钟 , 即系统时钟为 11.059 2 MHz。

系统时钟周期 :

$$t_{\text{sys}} / \text{ns} = 1/f_{\text{sys}} = 90.422454$$

传输 1 bit 需要的时间 :

$$T_{\text{bit}} / \text{ns} = 1 / (1 \text{ Mbps}) = 1000$$

包含的时间量子数 :

$$n = T_{\text{bit}} / t_{\text{sys}} = 11.0592$$

实际取 $n=11$, 从而实际波特率为

$$1 / (11 \times t_{\text{sys}}) = 1.005381818 \text{ Mbps}$$

则波特率误差为 0.5381%。

由以上可知, 由于四舍五入的原因同样会产生误差, 可通过选择适当的晶振频率减小该类误差。

综合考虑以上 2 种条件, 笔者选择的晶振的振荡误差范围为

$$\min(df, df) - \text{波特率误差} \quad (7)$$

选择的晶振的变化范围一定要在该范围之内, 否则可能导致通信失败。

4 波特率参数确定步骤及示例

下面结合 1 个具体例子来说明波特率的设置步骤及波特率与寄存器的操作关系。系统参数如表 2 所示。

表 2 系统参数表

参数	数值	说明
f_{sys}	11.0592 MHz	外部 22.1184 MHz 晶振经 2 分频整形得到
t_{sys}	90.422454 ns	$1/f_{\text{sys}}$
t_q	90.422454 ns	$t_{\text{sys}} \times BRRP$
CAN 总线长度	10 m	最远的 2 个 CAN 节点间距离
传输延时	400 ns	环路总延时

计算步骤如下 :

(1) 确定标称位时间 T_{bit} 包含的时间量子数 n , 同时计算波特率误差。该步骤计算过程见上面计算波特率误差计算过程。

(2) 确定传播时间段 ($Prop_seg$)。传播延时 = $2 \times (\text{线路传输延时} + \text{CAN 收发器延时})$, 其中线路延时为 5 ns/m, CAN 收发器的内部延时一般取 150 ns, 线路长度取最远的 2 个 CAN 节点间距离。如果 2 个节点间距离为 10 m, 则传播延时 = $2 \times (5 \text{ ns/m} \times 10 \text{ m} + 150 \text{ ns}) = 400 \text{ ns}$, 传播段包含的 t_q 为 $400 \text{ ns} / t_{\text{sys}} = 4.42$, 考虑总线冗余取其最大整数 5, 即 $Prop_seg = 5t_q$ 。

(3) 确定相位缓冲段 1 ($Phase_seg1$)、相位缓冲段 2 ($Phase_seg2$)、 SJW 的值。 $Sync_seg = 1t_q$ (固定), 剩下的 $11 - 5 - 1 = 5t_q$ 分配给 $Phase_seg1$ 和 $Phase_seg2$ 。这里取 $Phase_seg1 = 2t_q$ 、 $Phase_seg2 = 3t_q$, 最后 $SJW = \min(4, Phase_seg1) = 2t_q$ 。

(4) 计算容错范围 df 、确定晶振范围。

根据式(5)可得 :

$$df = \frac{\max(Phase_seg1, Phase_seg2)}{2 \times (13 \times T_{\text{bit}} - Phase_seg2)}$$

$$2 / [2 \times (13 \times 11 - 3)]$$

$$0.7246\%$$

根据式(6)可得 :

$$df = \frac{SJW}{20 \times T_{\text{bit}}}$$

$$2 / (20 \times 11)$$

$$0.9091\%$$

根据式(7)求得晶振振荡误差范围为

$$\min(0.7246\%, 0.9091\%) - 0.5381\% \\ = 0.1865\%$$

可见, 选择晶振的误差范围在 0.1865% 内就可满足上面的通信要求。

(5) 写波特率寄存器。

根据式(3)可以很方便地写波特率寄存器 :

波特率寄存器 = $BRRP + SJW \times 0x0040 +$

$$T_{\text{seg1}} \times 0x0100 + T_{\text{seg2}} \times 0x1000 \\ = 0x2640$$

将该值写入波特率寄存器就可得到要求的 1 Mbps CAN 总线通信速率。

5 结语

CAN 总线应用很多, 但是其参数的选择具有一定的难度, 需涉及到许多电气参数, 只有正确地理解这些值的作用和来源才能更好地掌握和应用 CAN 总线。在不同的系统应用中, 可以根据所使用的时钟频率、时钟信号的频率偏差、通信波特率及最大传输距离等因素, 对波特率寄存器参数进行优化确定。在选择晶振时需考虑晶振的振荡范围是否满足条件, 因为这关系到网络通信的性能, 甚至决定能否正常通信。笔者设计的某煤矿 CAN 总线系统目前已经投入使用, 运行可靠、稳定。

参考文献 :

- [1] 邬宽明. 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, 1997.
- [2] 董长飞. C8051F 系列单片机开发与 C 语言编程 [M]. 北京: 北京航空航天大学出版社, 2005.
- [3] HORTON L. C 语言入门经典 [M]. 4 版. 北京: 清华大学出版社, 2008.
- [4] 汪 瞳. CAN 总线网络仿真系统真系统开发 [D]. 哈尔滨: 哈尔滨工程大学, 2007.