第25卷第3期 武汉科技大学学报(自然科学版) Vol. 25, No. 3
2002年9月 J. of Wuhan Uni. of Sci. & Tech. (Natural Science Edition) Sep. 2002

挖掘机器人控制系统CAN总线接口

周 波[1],胡 骁[2],周 辉[3]

(1. 武汉科技大学,湖北 武汉,430081;2. 浙江大学,浙江 杭州,310027;3. 武汉钢铁(集团)公司,湖北 武汉,430083)

摘要:根据挖掘机器人控制系统的设计特点,采用基于CAN总线通信的多控制主体结构方案,实现了单片机CAN总线接口硬件和软件的设计。

关键词:CAN总线;挖掘机器人;多主控制

中图分类号:TP242.2 **文献标识码**:A **文章编号**:1001-4985(2002)03-0288-05

传统的机器人控制系统为集中式结构,这种结构处理器计算量大,可靠性差,维护困难[3]。分布式结构可靠性好,实时性强,易于维护[2]。近几年来,为了使机器人适应复杂的外界环境,能并行执行多种控制功能,多智能主体结构成为研究热点。这种控制系统结构的机器人由多个具备独立自主能力的控制单元组成,各控制单元之间通过实时总线或实时网络通信[3,4]。

液压挖掘机器人是一种重型可移动机械装置,工作环境恶劣,振动噪音大,对于控制系统的抗振动、抗干扰、可靠性和维护性能要求非常高。目前作者参与研制的挖掘机器人的发动机控制、液压系统控制、工作装置控制以及众多传感器都有其独立的控制器,各控制器之间只有少量的信息传输,需要实时控制总线连接各控制器,组成移动机器人的多主体控制系统。控制器局域网CAN总线特别适合于车辆和行走机械等工作环境恶劣的控制系统[1,2,3,6],既保证实时性,又能灵活增添规划或控制功能[3]。

1 液压挖掘机器人控制系统结构

液压挖掘机器人的控制功能模块具有视频监视力反馈无线遥控、柴油机-液压泵匹配节能控制、液压系统节能控制、在线工况监测、工作装置智能运动规划和决策、工作装置轨迹跟踪等功能,且均有相应独立的控制器。各控制器可以用CAN总线连接,如图1所示。这样的控制系统结构优点如下:①开放性好。每个控制器硬件和软件平台可以不同,如需要其他功能,可独立研究开发,然后加入;②实时性好。每一个控制器只执行少量的计算;③易维护。一个控制器需要更新或控制器失效,不影响其他控制器;④可靠性强。控制器之间连接导线少,受外界干扰机会少,而且CAN总线本身的抗干扰能力很强;⑤易安装。每个控制器体积较小,不但不受空间限制,而且还能根据传感器或执行器的位置就近安装,减小信号线长度。如果采用集中式结构,则控制器体积大,要选择合适的安装地点不容易;另外,造成连接传感器和执行器的信号线很长,且线路集结交错在一起,干扰机会增多。

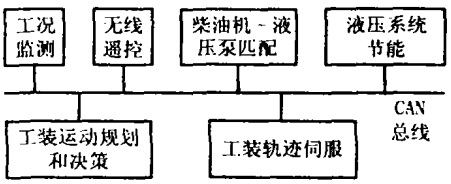

图1 控制系统结构

2 CAN总线接口设计

CAN芯片(或带CAN功能的芯片)型号多样,Intel, Philip, Motorola公司都有产品。一类是集成了CAN控制器的微处理器,如Philip公司的80C592等;另一类是独立的CAN控制器,如Philip公司的82C200, SJA1000(对82C200向下兼容),可作为一种微控制器的外围可编程芯片[5]。各种芯片使用方法不一,但所构成的基本CAN总线系统均有如下重要性能[6]:①CAN通信速率在40 m内可达1MB/s,距离达10km时速率为5KB/s,支持分布式控制或实时控制的串行通讯网络。②采用基于发送报文的编码方法,而不是传统的对各节点进行地址编码,可方便地实现系统的扩充或缩减;③为多主总线网络,即网络上任一节点都有权主动向其他节点传送数据;采用非破坏性

收稿日期:2002-01-26

基金项目:浙江省自然科学基金资助项目(699094),高等学校博士学科点专项科研基金资助项目(96033521).

作者简介:周波(1971—),男,武汉科技大学信息科学与工程学院,工程师,硕士.

总线优先级仲裁技术，通讯冲突时，优先级低的节点主动停发数据；采用 CRC 及其他检错纠错技术，因此总线冲突的解决可由 CAN 控制器自动完成，且能区分暂时和永久故障，自动关闭故障节点；④具有数据帧、远程帧、出错帧、超载帧四种报文帧格式。采用短帧结构，每帧有效字节数为 8 个，因此传输时间短，受干扰的概率低，支持高可靠的实时控制。

图 1 中，有的模块采用单片机 80C552（8051 系列）控制，有的由工业 PC 机控制。80C552 采用 SJA1000 作 CAN 控制器，82C250 作 SJA1000 与物理总线的接口芯片[5]。PC 机上选用研华公司的 PCL-841 双端口 CAN 接口卡，接口卡内部芯片也是 SJA1000 和 82C250。

2.1　PC 机 CAN 总线通信接口适配卡 PCL-841

PCL-841 插在 PC 机的扩展槽内，有两个 CAN 接口。PC 机能以访问内存的方式读写适配卡上 CAN 控制器的寄存器单元。PCL-841 提供丰富的使用 CAN 总线的 C 语言库函数，且提供源代码例程和试验工具软件，在进行 CAN 总线接口编程时可以直接将这些源代码例程移植到自己的程序中，提高了开发效率，这里不详述。通过跳线和 DIP 拨盘开关的设置，可使两个 CAN 接口拥有自己的中断号和内存映射地址区。

2.2　单片机和 CAN 控制器 SJA1000 及其与物理总线的接口设计

CAN 接口控制器与单片机硬件接口原理如图 2 所示。用地址译码器访问 SJA1000 芯片，而

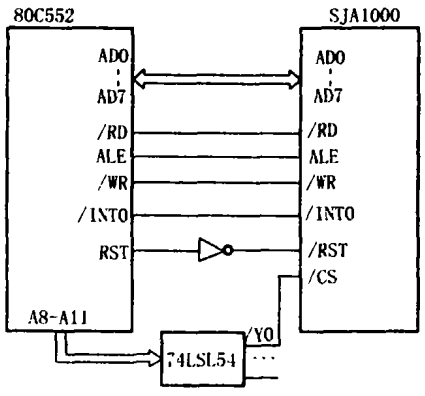

图 2　CAN 控制器接口原理

且将 SJA1000 的各内部寄存器映射到单片机外部 RAM 的某段高端地址区域，程序则以读取外部 RAM 的方式访问 CAN 芯片。SJA1000 的/CS 信号端被 4～16 译码器 74LS154 的片选信号/Y0 选中时，即被 80C552 访问，此时 SJA1000 的首地址即为 8000H。读写（/RD /WR）、复位（RST）、中断（/INT0）、地址锁存（ALE）信号线均直接与 80C552 连接。80C552 与 CAN 控制器的接口硬件完整设计见图 3。

CAN 接口控制器软件设计主要工作是详细定义控制段寄存器。下面是已经调试成功的试验程序（C51 编制），它可完成 PC 机和单片机之间数据传送。单片机采用中断法收发 PC 机的数据，在收到命令帧后回送给 PC 机一帧数据。

```
#include < reg552. h > /* 80C552 的 C51 语言头文件 */
/* CAN 控制器 SJA1000 内部寄存器的地址分配 8000H-8031H，0x28000L 中 2 指外部 RAM，8000H 是 SJA1000 内部寄存器起始地址 */
#define XBYTE ((unsigned char volatile xdata *) 0x28000L )
/* 以下定义 SJA1000 的 10 个控制寄存器 */
/ * 控制寄存器物理地址为 8000H */
#define ModeControlReg XBYTE[0]
#define CommandReg XBYTE[1]/ * 命令寄存器 */
#define StatusReg XBYTE[2]/ * 状态寄存器 */
#define InterruptReg XBYTE[3]/ * 中断寄存器 */
/ * 验收码寄存器 */
#define AcceptCodeReg XBYTE[4]
/ * 验收屏蔽寄存器 */
#define AcceptMaskReg XBYTE[5]
/ * 总线定时 0 寄存器 */
#define BusTiming0Reg XBYTE[6]
/ * 总线定时 1 寄存器 */
#define BusTiming1Reg XBYTE[7]
/ * 输出控制寄存器 */
#define OutControlReg XBYTE[8]
/ * 发送缓冲区的地址定义，共 10 字节 */
#define TxBuffer1 XBYTE[10]
#define TxBuffer2 XBYTE[11]
#define TxBuffer3 XBYTE[12]
#define TxBuffer4 XBYTE[13]
#define TxBuffer5 XBYTE[14]
#define TxBuffer6 XBYTE[15]
#define TxBuffer7 XBYTE[16]
#define TxBuffer8 XBYTE[17]
#define TxBuffer9 XBYTE[18]
#define TxBuffer10 XBYTE[19]
/ * 定义接收缓冲区的十个寄存器 */
```

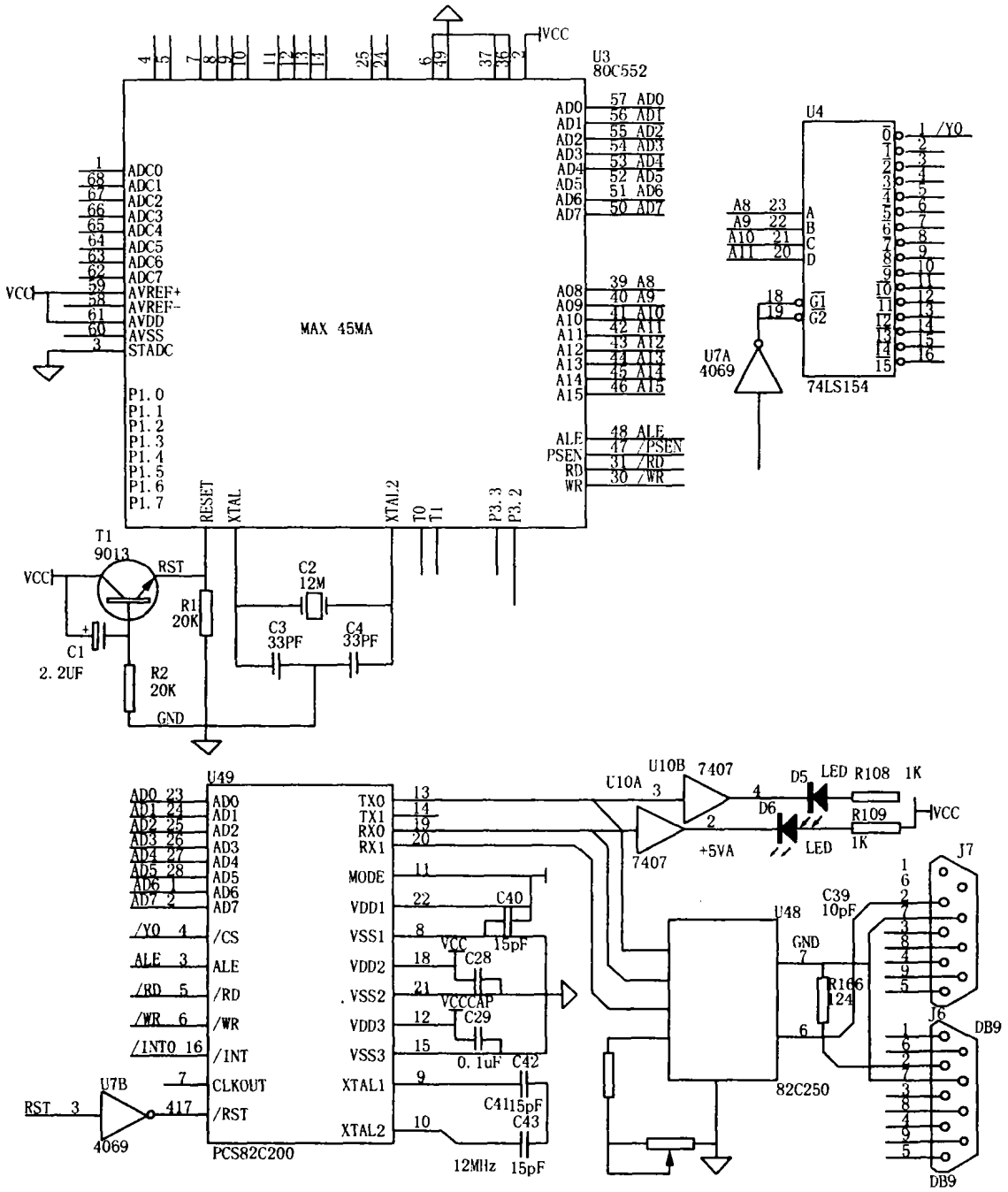

图 3 单片机 80C552 的 CAN 接口的硬件设计

```
#define RxBuffer1 XBYTE[20]
#define RxBuffer2 XBYTE[21]
#define RxBuffer3 XBYTE[22]
#define RxBuffer4 XBYTE[23]
#define RxBuffer5 XBYTE[24]
#define RxBuffer6 XBYTE[25]
#define RxBuffer7 XBYTE[26]
#define RxBuffer8 XBYTE[27]
#define RxBuffer9 XBYTE[28]
#define RxBuffer10 XBYTE[29]
/*时钟分频寄存器*/
#define ClockDivideReg XBYTE[31]
```

```
/*下面是数值及标志位的宏定义*/
/* PrescExample , SJWExample , TSEG 1 Example 和 TSEG2Example 用于决定 CAN 总线数据传输波特率*/
#define PrescExample 0x02 /*波特率预分频*/
#define SJWExample 0xC0 /*同步跳转宽度*/
/*TSEG1Example 和 TSEG2Example 确定传输数据时的位周期宽度,采样点位置*/
#define TSEG1Example 0x0A
#define TSEG2Example 0x30
```

```
#define TBS_Bit 0x04 /* 允许向发送缓冲区写入报文的标志位 */
#define TR_Bit 0x01 /* 发送请求置位位 */
bit flag ;
unsigned char status ;
/* 80C552 中断 0 的服务例程 */
void Service_TransmitInt( ) interrupt 0 using 2
{
 * 接收缓冲区 RxBuffer1 - RxBuffer10 中已有的 PC 机传来的数据,程序可直接调用这些数据 */
/* 下面开始向 PC 机发送一帧数据 */
CommandReg = TBS_Bit ;/* 发送请求位置位 */
If ( InterruptReg&0x01 = =0x01)
{
StatusReg =0x04; /* 向上位机发送一帧数据 */
TxBuffer1 = 0xA5; /* 帧标志符 */
TxBuffer2 = 0x28;
TxBuffer3 = 0x01; /* 第 1 字节 (仅示例) */
TxBuffer4 = 0x02;
TxBuffer5 = 0x03;
TxBuffer6 = 0x04;
TxBuffer7 = 0x05;
TxBuffer8 = 0x06;
TxBuffer9 = 0x07;
TxBuffer10 = 0x08; /* 第 8 字节 */
CommandReg = TR_Bit ; /* 启动发送 */
}
}
void main(void) /* 主程序开始 */
{
/* SJA1000 初始化 */
PX0 = 1 ;/* SJA1000 中断为高优先级 */
IT0 = 0 ; /* 电平触发方式引起中断 */
EA = 0 ; /* 屏蔽 80C552 所有中断源 */
EX0 = 0; /* 禁止 SJA1000 向 80C552 请求中断 */
ModeControlReg = 0x01; /* SJA1000 暂时禁止所有中断,并请求复位 */
ClockDivideReg = 0x48; /* 根据上图硬件线路设定输出方式:SJA1000 芯片 TX1 悬空,TX0 发送数据;BasicCAN 工作模式 */
/* AcceptCodeReg 和 AcceptMaskReg 决定 SJA1000 能接收到哪一个节点的数据帧,在这里,试验没有屏蔽外来数据,允许接收所有外来的数据帧 */
AcceptCodeReg = 0x00;
AcceptMaskReg = 0xFF;
/* 波特率为 125Kbps */
BusTiming0Reg = SJWExample | PrescExample;
BusTiming 1 Reg = TSEG 2 Example | TSEG 1 Example;
ModeControlReg = 0x02; /* 接收中断开放 */
EX0 = 1; /* 允许 SJA1000 请求中断,返回正常运行状态 */
EA = 1; /* 开放所有中断源 */
/* 初始化完后,单片机开始运行自己的程序 */
/* 但在试验中,我们用无限循环等待上位机传来的数据 */
for( ; ;) { }
} /* 主程序终止 */
```

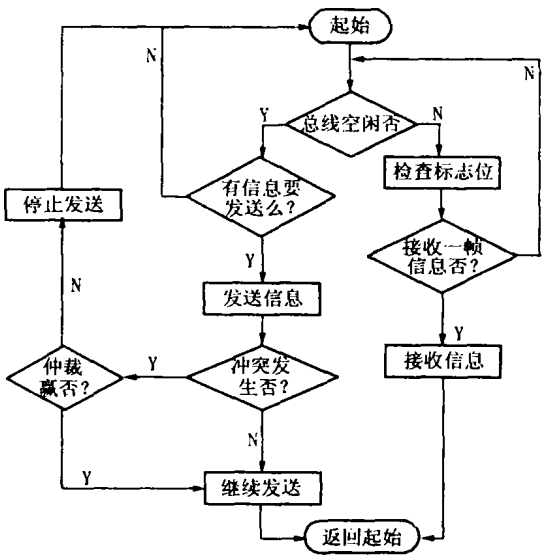

图 4　CAN 总线信息冲突仲裁流程

3.3　信息传输冲突

CAN 总线上的独立控制器很多,控制器只有在总线空闲时才能传输数据。有时总线正忙,多个控制器同时传输或接收数据,因而存在一个信息传输冲突仲裁问题,CAN 控制器通过比较各帧数据的帧标志符解决该问题。帧标志符在每帧数据的第一字节。仲裁过程见图 4。一帧数据传输完毕,其他各帧数据再接受仲裁以决定谁先发送。

参 考 文 献

[1] Lee Jang Myung, Lee Suk, Lee Man Hyung, et al. Integrated Wiring System for Construction Equipment[A]. IEEE/ASME Transactions on Mechatronics [C]. 1999,4:187—195.

[2] Wurll C, Henrich D, Worn H, et al. A Distributed Planning and Control System for Industrial Robots [A]. 5th International Workshop on Advanced Motion Control[C]. 1998: 487—492.

[3] Wargui M, Rachid A. Application of Controller Area Network to Mobile Robots [A]. 8th Mediterranean Electrotechnical Conference[C]. 1996,1:205—207.

[4] Hasnaoui S, Bouallegue A. Influence of Multipath Fading on the Implementation of a Radio Sensor as an Isolated Station on a Web CAN-based Control Systems [A]. Proceedings of the 16th IEEE on Instrumentation and Measurement Technology Conference[C]. 24—26 May 1999,1:475—480.

[5] Application Note—SJA1000-Stand-alone CAN controller AN97076[Z]. Philips Electronics N. V. ,1997.

[6] 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,1996.

CAN Interface for the Control System of an Excavating Robot

ZHOU Bo[1], *HU Xiao*[2], *ZHOU Hui*[3]

(1. Wuhan University of Science and Technology, Wuhan 430081, China; 2. Zhejiang University, Hangzhou 310027, China; 3. Wuhan Iron and Steel Corporation, Wuhan 430083, China)

Abstract: Based on the features of the control system design of the excavating robot, this paper employs CAN bus-based multi-host control structure and has realized the implementation of the hardware and software design of CAN interface in microchip computer.

Keywords: CAN bus; excavating robot; multi-host control

[责任编辑 徐前进]

·科技快讯·

揭开惠更斯复摆钟之谜

研究荷兰杰出物理学家惠更斯著作的科学家,终于成功地揭开原先一直难以理解的 17 世纪复摆钟之谜。众所周知,惠更斯在 1657 年发明了摆钟,试图解决测定海洋经度的问题,提出了钟摆理论并创立了复摆摆动定律。在研究过程中,惠更斯设计了两对钟,如果其中一对钟停摆或需要修理和清洗,第二对钟可以继续不间断地工作。

在研究初期,惠更斯发现伽利略确信的关于摆锤振动的等时性并不准确,摆锤振动的等时性只有在小角度偏离垂线时才成立,但是在 6 暇偏角范围内摆锤振动明显没有等时性。惠更斯在 1673 年发现,偏角为 90 度时的周期性与 34—29 度较小偏角时的周期性相似。为了补偿偏离的等时性,惠更斯决定在增大偏角时编短摆锤的长度。在最初制作的这些摆钟里,他在曲柄状限止器上部分地绕上悬线。选择曲柄状的方法并没有使惠更斯满意,1658 年,他从摆钟中去除了曲柄状限止器,而加入摆幅限止器,但是这并不表示他放弃寻找摆锤的等时性。在重 659 年制作的摆钟里又重新出现了曲柄状限止器,不过这一次惠更斯已经能从理论上确定曲柄限止器的形状。

当惠更斯释放摆锤时,他发现了某种奇怪现象尸—后来他称之为"奇怪的相互理解",无论他怎样释放摆锤,最终它们都会朝相反方向转动。直到不久前,美国亚特兰大技术学院研究人员库尔特·瓦辛费尔德首先对这一问题进行试验,重新制作了惠更斯摆钟的复制品,并且仔细分析它们的走动。瓦辛费尔德发现了类似的等时性,但是这种情况只有在摆锤重量比整个摆钟结构重量轻很多时才会发生。如每个摆锤的质量与整个摆钟结构质量之比小于 1:120,则摆锤就会开始朝相反方向转动。如果这一比值大于重:80,则其中一个摆锤或两个摆锤会逐渐停摆。

库尔特·瓦辛费尔德认为,长期以来人们之所以无法揭开惠更斯摆钟之谜,是因为直到今天才对所谓的非线性问题感兴趣。 (摘自《世界发明》2002 年第 7 期,周道其/文)