



(12) 发明专利

(10) 授权公告号 CN 112114982 B

(45) 授权公告日 2022.03.08

(21) 申请号 202010941157.X

(22) 申请日 2020.09.09

(65) 同一申请的已公布的文献号  
申请公布号 CN 112114982 A

(43) 申请公布日 2020.12.22

(73) 专利权人 天津津航计算技术研究所  
地址 300308 天津市东丽区空港经济区保  
税路357号

(72) 发明人 王青松 李方 所玉君 崔建飞

(74) 专利代理机构 中国兵器工业集团公司专利  
中心 11011  
代理人 王雪芬

(51) Int. Cl.  
G06F 9/54 (2006.01)

(56) 对比文件

- CN 1504892 A, 2004.06.16
- CN 103019823 A, 2013.04.03
- CN 110018914 A, 2019.07.16
- CN 107992370 A, 2018.05.04
- CN 110764935 A, 2020.02.07
- US 2011078700 A1, 2011.03.31

审查员 李金蔓

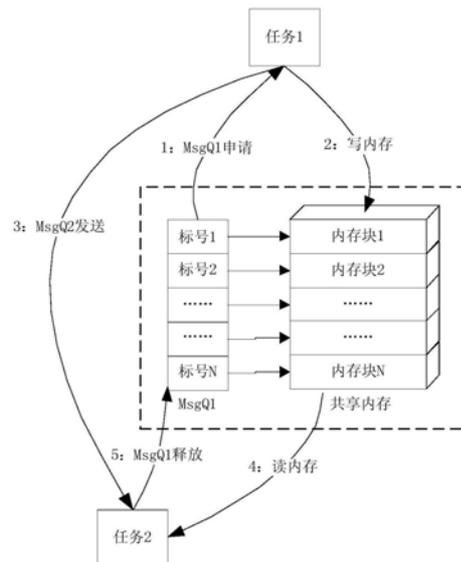
权利要求书1页 说明书4页 附图1页

(54) 发明名称

基于VxWorks系统的多任务间共享内存的管理方法

(57) 摘要

本发明涉及一种基于VxWorks系统的多任务间共享内存的管理方法,属于VxWorks共享内存技术领域。本发明首先根据实际需要动态申请一块全局内存,并将内存块等长划分为N个数据块,建立M+1个成员数为N的全局消息队列,并用数字1~N标号将消息队列1填满,标号与全局共享内存的数据块一一对应,多个任务根据消息队列1得到标号,来对共享内存进行操作,并通过消息队列2~M+1对接收任务1~M进行告知,从而高效地实现多任务间共享内存的管理和数据通信。本发明通过通用化的设计,解决了多任务间频繁大量数据交换的需求,并提高了程序的可靠性,适用于所有使用VxWorks系统的软件。



1. 一种基于VxWorks系统的多任务间共享内存的管理方法,其特征在于,包括以下步骤:

步骤1、确定总的执行任务,包括数据接收任务、数据记录任务、断电保护任务、周期自检任务,设置任务优先级顺序为:数据接收任务>断电保护任务>数据记录任务>周期自检任务,其中数据记录任务的数量为1;

步骤2、根据估算的实际通信容量和频率,通过malloc函数动态申请一块容量足够的全局内存g\_pool,并将内存块g\_pool等长划分为N个数据块,其中根据接收的数据量将每个数据块的大小定为1024字节,数据块的个数定为10000;

步骤3、通过msgQCreate函数先建立1个成员数为10000的全局消息队列,命名为MsgQ1,将其属性设置为先进先出MSG\_Q\_FIFO,数据长度为2字节即可满足标号为10000的需求,并将数字1~10000发送至队列MsgQ1,将MsgQ1填满,通过此方式将MsgQ1的标号与共享内存g\_pool中的每个数据块建立一一对应的关系;然后再建立1个成员数为10000的全局消息队列,命名为MsgQ2,将其属性设置为按优先级排队MSG\_Q\_PRIORITY,数据长度为6字节,包括消息标号、消息类型、消息长度三个信息;

步骤4、在系统启动后,数据接收任务通过接收MsgQ1申请标号,来对共享内存g\_pool进行操作,将共享内存g\_pool的数据填写完成后,通过发送消息至MsgQ2,完成对数据记录任务的告知;

步骤5、数据记录任务通过实时接收MsgQ2中信息,根据消息标号和数据长度来获取共享内存g\_pool中的数据内容,根据消息类型对数据内容进行解析及实时处理,并将接收到的标号发送回MsgQ1中;

步骤6、重复步骤4和5,数据接收任务、断电保护任务、周期自检任务在实时申请标号后,向数据记录任务发送数据,数据记录任务根据接收的消息执行对应操作,并完成对标号资源的释放。

2. 如权利要求1所述的方法,其特征在于,步骤4中,设数据接收任务发送消息类型为1~4,分别作为接收通道的标识。

3. 如权利要求2所述的方法,其特征在于,步骤4中,设断电保护任务发送的消息类型为5。

4. 如权利要求3所述的方法,其特征在于,步骤4中,设周期自检任务的消息类型为6。

5. 如权利要求4所述的方法,其特征在于,步骤4中,数据长度为实际传送数据长度,范围是1~1024字节。

6. 如权利要求5所述的方法,其特征在于,步骤4中,数据接收任务、周期自检任务发送MsgQ2时,参数选择正常优先级MSG\_PRI\_NORMAL发送,断电保护任务发送MsgQ2时,参数选择MSG\_PRI\_URGENT发送。

7. 如权利要求6所述的方法,其特征在于,步骤5中,数据记录任务对于消息类型为1~4的数据内容,分通道进行数据存储记录。

8. 如权利要求7所述的方法,其特征在于,步骤5中,若接收到的消息类型为5,代表检测到了断电信号,此时停止写数据文件的操作。

9. 如权利要求8所述的方法,其特征在于,步骤5中,对于类型为6的消息,单独存储或与其中一个通道合并存储。

## 基于VxWorks系统的多任务间共享内存的管理方法

### 技术领域

[0001] 本发明属于VxWorks共享内存技术领域,具体涉及一种基于VxWorks系统的多任务间共享内存的管理方法。

### 背景技术

[0002] VxWorks是由美国风河公司推出的运行在目标机上的高性能、可裁减的嵌入式实时操作系统。VxWorks以其卓越的实时性和良好的可靠性被广泛地应用在通信、军事、航空、航天等高精尖技术及实时性要求高的领域中。

[0003] 为了保证整机的实时性能,目前稍复杂的程序开发大多采用多任务设计,将实时性能要求最高的任务分配最高优先级,其他任务在满足功能性能指标的同时可以延缓执行,多任务之间往往需要进行交互,目前主要存在以下几种通信方式。

[0004] 1、信号量是任务间最快速的通信机制,VxWorks系统提供了二进制、互斥和计数三种信号量,用于解决任务间的互斥和同步。二进制信号量主要用于任务的同步,互斥信号量可以解决内在的互斥问题,主要用于优先级继承、安全删除等,计数信号量在二进制信号量的基础上增加了计数功能,可以解决一个资源多个成员需要保护的情况。信号量机制通信速度快、系统开销小、处理方法简单,但无法附带更多的信息,不能用于数据交换。

[0005] 2、在VxWorks中,消息队列是一种相对高级的任务间通信机制,消息队列的使用灵活,实现起来比信号量稍复杂。优点是它可以传递更多的信息,能够用于任务间的数据通信,但由于它的系统开销较高,当传送消息频繁或数据量大时,通信效率会降低,这对于高实时性系统环境来说,可能导致通信超时问题。

[0006] 3、共享内存是任务间数据通信常用的方法,在多任务间通信时将同一块地址空间的内存进行共享,为防止访问冲突,目前常见的做法是用互斥信号量对共享内存进行上锁,保证数据的正确性。该方法不允许多个任务同时访问该区域,操作上比较安全,但对于需要大量数据交换的情况,该方法效率低下,可能存在优先级高的任务过度抢占,导致低优先级任务无法正常工作。

[0007] 为了弥补现有多任务间数据通信处理方法的不足,在满足任务间大数据量频繁通信的需求的同时,提高程序的响应实时性和运行可靠性,迫切需要一种更优的解决方案。

### 发明内容

[0008] (一)要解决的技术问题

[0009] 本发明要解决的技术问题是:如何实现一种基于VxWorks系统的多任务间共享内存管理方法,解决现有多任务间数据通信处理方法的问题,满足任务间大数据量频繁通信需求的同时,提高程序的响应实时性和运行可靠性。

[0010] (二)技术方案

[0011] 为了解决上述技术问题,本发明提供了一种基于VxWorks系统的多任务间共享内存的管理方法,包括以下步骤:

[0012] 步骤1、确定总的执行任务,包括数据接收任务、数据记录任务、断电保护任务、周期自检任务,设置任务优先级顺序为:数据接收任务>断电保护任务>数据记录任务>周期自检任务,其中数据记录任务的数量为1;

[0013] 步骤2、根据估算的实际通信容量和频率,通过malloc函数动态申请一块容量足够的全局内存g\_pool,并将内存块g\_pool等长划分为N个数据块,其中根据接收的数据量将每个数据块的大小定为1024字节,数据块的个数定为10000;

[0014] 步骤3、通过msgQCreate函数先建立1个成员数为10000的全局消息队列,命名为MsgQ1,将其属性设置为先进先出MSG\_Q\_FIFO,数据长度为2字节即可满足标号为10000的需求,并将数字1~10000发送至队列MsgQ1,将MsgQ1填满,通过此方式将MsgQ1的标号与共享内存g\_pool中的每个数据块建立一一对应的关系;然后再建立1个成员数为10000的全局消息队列,命名为MsgQ2,将其属性设置为按优先级排队MSG\_Q\_PRIORITY,数据长度为6字节,包括消息标号、消息类型、消息长度三个信息;

[0015] 步骤4、在系统启动后,数据接收任务通过接收MsgQ1申请标号,来对共享内存g\_pool进行操作,将共享内存g\_pool的数据填写完成后,通过发送消息至MsgQ2,完成对数据记录任务的告知;

[0016] 步骤5、数据记录任务通过实时接收MsgQ2中信息,根据消息标号和数据长度来获取共享内存g\_pool中的数据内容,根据消息类型对数据内容进行解析及实时处理,并将接收到的标号发送回MsgQ1中;

[0017] 步骤6、重复步骤4和5,数据接收任务、断电保护任务、周期自检任务在实时申请标号后,向数据记录任务发送数据,数据记录任务根据接收的消息执行对应操作,并完成对标号资源的释放。

[0018] 优选地,步骤4中,设数据接收任务发送消息类型为1~4,分别作为接收通道的标识。

[0019] 优选地,步骤4中,设断电保护任务发送的消息类型为5。

[0020] 优选地,步骤4中,设周期自检任务的消息类型为6。

[0021] 优选地,步骤4中,数据长度为实际传送数据长度,范围是1~1024字节。

[0022] 优选地,步骤4中,数据接收任务、周期自检任务发送MsgQ2时,参数选择正常优先级MSG\_PRI\_NORMAL发送,断电保护任务发送MsgQ2时,参数选择MSG\_PRI\_URGENT发送。

[0023] 优选地,步骤5中,数据记录任务对于消息类型为1~4的数据内容,分通道进行数据存储记录。

[0024] 优选地,步骤5中,若接收到的消息类型为5,代表检测到了断电信号,此时停止写数据文件的操作。

[0025] 优选地,步骤5中,对于类型为6的消息,单独存储或与其中一个通道合并存储。

[0026] 本发明还提供了一种所述的方法在VxWorks共享内存技术领域中的应用。

[0027] (三)有益效果

[0028] 本发明首先根据实际需要动态申请一块全局内存,并将内存块等长划分为N个数据块,建立M+1个成员数为N的全局消息队列,并用数字1~N标号将消息队列1填满,标号与全局共享内存的数据块一一对应,多个任务根据消息队列1得到标号,来对共享内存进行操作,并通过消息队列2~M+1对接收任务1~M进行告知,从而高效地实现多任务间共享内存

的管理和数据通信。本发明通过通用化的设计,解决了多任务间频繁大量数据交换的需求,并提高了程序的可靠性,适用于所有使用VxWorks系统的软件。

### 附图说明

[0029] 图1是本发明一种基于VxWorks系统的多任务间共享内存的管理方法中通信原理图。

### 具体实施方式

[0030] 为使本发明的目的、内容和优点更加清楚,下面结合附图和实施例,对本发明的具体实施方式作进一步详细描述。

[0031] 本发明为VxWorks系统多任务间频繁大量数据交换的需求,提供了一种通用的解决方案,该方案能够支持多个任务对共享内存同时进行操作,并完成任意任务之间的大量数据交换,在保证任务间数据处理的实时性的同时,提高了系统可靠性。本发明采用通用化的设计,逻辑机理清晰,实现灵活简单,程序可移植性高,适用于所有使用VxWorks系统的软件。

[0032] 本发明一种基于VxWorks系统的多任务间共享内存的管理方法的技术原理是将消息队列1至消息队列M+1共M+1个消息队列中的消息队列1和共享内存块进行绑定,两者数量一一对应,利用VxWorks系统对消息队列的维护机制,保证多任务间能够同时通过接收消息队列1来申请利用共享内存块,并通过消息队列2~M+1来将共享数据传递给其他任务。以最简单的一发一收任务模型为例,该数据流原理如图1所示。

[0033] 进一步地,该消息队列1的功能相当于一组信号量,消息队列1中的每个成员相当于一个信号量,成员内存放的是共享内存块的标号,得到标号才可以访问对应的共享内存块,从而在系统层面防止了多个任务间对共享内存的访问冲突,提高了程序运行的可靠性。

[0034] 进一步地,该消息队列2对应其中任务1的数据接收,其他多个任务均可以通过该消息队列2向任务1发送消息,消息队列2中存放的内容为共享内存块的标号、数据类型和实际有效内容的长度,仅占用几个字节即可进行收发任务之间的信息传递,而实际对应共享内存块中可能包含几KB的实际内容,通过类似指针处理的方式,可以大幅提高任务间数据传递的效率。

[0035] 进一步地,消息队列2的与任务间的映射关系为多对1,用于任务间通信的消息队列的数量等于需要接收数据的任务数量,从而能够实现任务间的任意通信。

[0036] 下面结合具体的实施例,将上述方案进行详细分解,某任务数据记录器要保证接收数据实时性、完整性的同时,完成设备状态上报,及数据记录的功能。具体实施步骤如下:

[0037] 步骤1、首先,根据系统需求确定总的执行任务,包括数据接收任务、数据记录任务、断电保护任务、周期自检任务,设置任务优先级顺序为:数据接收任务>断电保护任务>数据记录任务>周期自检任务,其中,数据接收任务、断电保护任务、周期自检任务对应图1中的任务1,数据记录任务对应图1中的任务2,其数量为1;

[0038] 步骤2、其次,根据估算的实际通信容量和频率,通过malloc函数动态申请一块容量足够的全局内存g\_pool,并将内存块g\_pool等长划分为N个数据块。这里,根据接收的数据量将每个数据块的大小定为1024字节,数据块的个数定为10000;

[0039] 步骤3、通过msgQCreate函数先建立1个成员数为10000的全局消息队列,命名为MsgQ1,将其属性设置为先进先出MSG\_Q\_FIFO,数据长度为2字节即可满足标号为10000的需求,并将数字1~10000发送至队列MsgQ1,将MsgQ1填满,通过此方式将MsgQ1的标号与共享内存g\_pool中的每个数据块建立一一对应的关系;然后再建立1个成员数为10000的全局消息队列,命名为MsgQ2,将其属性设置为按优先级排队MSG\_Q\_PRIORITY,数据长度为6字节,包括消息标号、消息类型、消息长度三个信息;

[0040] 步骤4、在系统启动后,数据接收任务通过接收MsgQ1申请标号,来对共享内存g\_pool进行操作,将共享内存g\_pool的数据填写完成后,通过发送消息至MsgQ2,完成对数据记录任务的告知。

[0041] 其中,数据接收任务发送消息类型为1~4,分别作为接收通道的标识;断电保护任务发送的消息类型为5,周期自检任务的消息类型为6;数据长度为实际传送数据长度,范围是1~1024字节。

[0042] 其中,数据接收任务、周期自检任务发送MsgQ2时,参数选择正常优先级MSG\_PRI\_NORMAL发送,断电保护任务发送MsgQ2时,参数选择MSG\_PRI\_URGENT发送,既保证正常接收数据的先进先出特性,又保证断电保护操作被执行的实时性。

[0043] 步骤5、数据记录任务通过实时接收MsgQ2中信息,根据消息标号和数据长度来获取共享内存g\_pool中的数据内容,根据消息类型对数据内容进行解析及实时处理,并将接收到的标号发送回MsgQ1中,这样其他任务(数据接收任务、断电保护任务、周期自检任务)能够在后续申请到该标号,完成了该内存块的回收。

[0044] 其中,数据记录任务对于消息类型为1~4的数据内容,分通道进行数据存储记录;若接收到的消息类型为5,代表检测到了断电信号,此时停止写数据文件的操作,防止文件意外损坏;对于类型为6的消息,可考虑单独存储或与其中一个通道合并存储。

[0045] 步骤6、重复步骤4和5,三个任务(数据接收任务、断电保护任务、周期自检任务)在实时申请标号后,向数据记录任务发送数据,数据记录任务根据接收的消息执行对应操作,并完成对标号资源的释放。

[0046] 通过上述步骤,稳定高效地实现了VxWorks系统多任务间共享内存的管理和数据通信。并且针对不同的系统需求,根据需缓存内容的大小,只需调整几个参数就能灵活地改变共享内存的大小、内存块数量和消息传输的格式,在保证程序可靠性和效率的同时,提高了程序的通用性和可移植性。

[0047] 以上所述仅是本发明的优选实施方式,应当指出,对于本技术领域的普通技术人员来说,在不脱离本发明技术原理的前提下,还可以做出若干改进和变形,这些改进和变形也应视为本发明的保护范围。

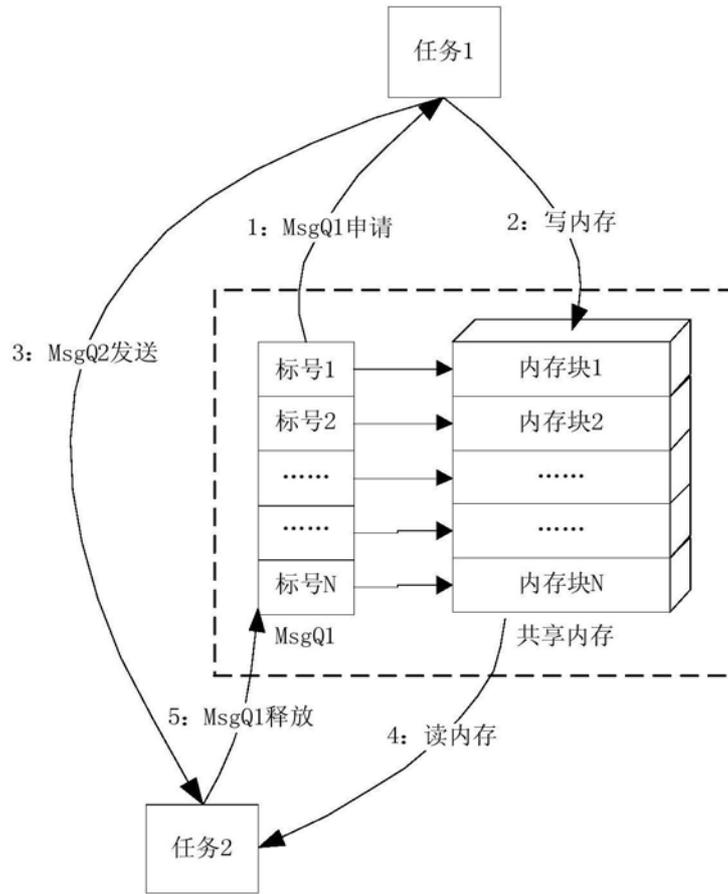


图1