



(12) 发明专利

(10) 授权公告号 CN 103257670 B

(45) 授权公告日 2016. 06. 01

(21) 申请号 201210041592. 2

JP H07219673 A, 1995. 08. 18,

(22) 申请日 2012. 02. 21

CN 101043253 A, 2007. 09. 26,

(73) 专利权人 北京融合视讯科技有限公司

审查员 刘志军

地址 100015 北京市朝阳区酒仙桥中路 24 号 878 东区 10 层

(72) 发明人 袁堂夫 邹义生

(74) 专利代理机构 北京银龙知识产权代理有限公司 11243

代理人 许静 赵爱军

(51) Int. Cl.

G06F 1/14(2006. 01)

(56) 对比文件

CN 101546169 A, 2009. 09. 30,

US 2004236976 A1, 2004. 11. 25,

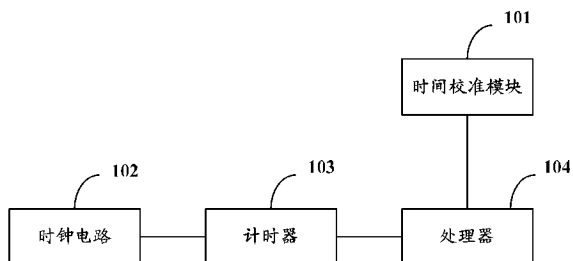
权利要求书2页 说明书9页 附图4页

(54) 发明名称

嵌入式系统及其计时方法

(57) 摘要

本发明提供一种嵌入式系统及其计时方法, 所述嵌入式系统包括: 时间校准模块, 用于获取系统的校准时间; 时钟电路, 用于产生时钟信号; 计时器, 用于接收所述时钟信号, 并根据接收到的所述时钟信号的数量进行计数, 得到一计数数值, 在所述计数数值达到预设数值时, 生成一中断事件通知, 并获取所述计数数值达到预设数值所需的时长; 处理器, 用于在接收到所述中断事件通知后, 执行中断服务程序, 所述执行中断服务程序包括: 统计所述中断服务程序的执行次数, 并根据所述时长、所述执行次数以及所述校准时间, 计算系统当前时间。本发明能够在不增加实时时钟电路的基础上实现嵌入式系统的精确计时, 并可避免处理器执行程序指令用时所带来的时间累积误差。



1. 一种嵌入式系统,其特征在于,包括:

时间校准模块,用于获取系统的校准时间;

时钟电路,用于产生时钟信号;

计时器,用于接收所述时钟信号,并根据接收到的所述时钟信号的数量进行计数,得到一计数数值,在所述计数数值达到预设数值时,生成一中断事件通知,并获取所述计数数值达到预设数值所需的时长;

处理器,用于在接收到所述中断事件通知后,执行中断服务程序,所述执行中断服务程序包括:统计所述中断服务程序的执行次数,并根据所述时长、所述执行次数以及所述校准时间,计算系统当前时间;

所述计时器还包括:

第一计时模块,用于在每接收到一所述时钟信号时,将所述计数数值加一,并获取所述计数数值从所述计时器的初始值递增计数到第一预设数值所需的时长;或

第二计时模块,用于在每接收到一所述时钟信号时,将所述计数数值减一,并获取所述计数数值从所述初始值递减计数到第二预设数值所需的时长;

所述处理器还包括:

计时器重置模块,用于在执行所述中断服务程序时,重置所述计时器的初始值;

误差获取模块,用于获取从接收到所述中断事件通知至重置所述计时器的初始值之间所述计时器的计数数值,作为误差值,并计算所述误差值与所述初始值的和值;

其中,所述计时器重置模块,还用于将所述和值作为所述计时器的初始值,并重置所述计时器的初始值;所述计数数值包括高位计数段和低位计数段,所述将所述和值作为所述计时器的初始值,并重置所述计时器的初始值具体为:将所述和值作为所述计时器的初始值,仅重置所述计时器的初始值的高位计数段的值。

2. 一种嵌入式系统的计时方法,其特征在于,包括:

接收时钟电路产生的时钟信号,并采用计时器根据接收到的所述时钟信号的数量进行计数,得到一计数数值,在所述计数数值达到预设数值时,生成一中断事件通知;

获取所述计数数值达到预设数值所需的时长;

获取系统的校准时间;

在接收到所述中断事件通知后,执行中断服务程序,所述执行中断服务程序包括:统计所述中断服务程序的执行次数,并根据所述时长、所述执行次数以及所述校准时间,计算系统当前时间;

所述根据接收到的所述时钟信号的数量进行计数,得到一计数数值的步骤包括:

在每接收到一所述时钟信号时,将所述计数数值加一,并获取所述计数数值从所述计时器的初始值递增计数到第一预设数值所需的时长;或

在每接收到一所述时钟信号时,将所述计数数值减一,并获取所述计数数值从所述初始值递减计数到第二预设数值所需的时长;

所述执行中断服务程序的步骤还包括:

重置所述计时器的初始值;

所述重置所述计时器的初始值的步骤包括:

获取从接收到所述中断事件通知至重置所述计时器的初始值之间的所述计时器的计

数数值,作为误差值,并计算所述误差值与所述初始值的和值;

将所述和值作为所述计时器的初始值,并重置所述计时器的初始值;

所述计数数值包括高位计数段和低位计数段,所述将所述和值作为所述计时器的初始值,并重置所述计时器的初始值的步骤包括:

将所述和值作为所述计时器的初始值,仅重置所述计时器的初始值的高位计数段的值。

3.如权利要求2所述的嵌入式系统的计时方法,其特征在于:

所述计数数值从所述初始值递增计数到第一预设数值所需的时长的计算公式如下:

$$\delta t = (L1 - pri) * T$$

其中, $\delta t$ 为计数数值从所述初始值递增计数到第一预设数值所需的时长, $pri$ 为所述初始值, $L1$ 为所述第一预设阈值, $T$ 为时钟周期;

所述计数数值从所述初始值递减计数到第二预设数值所需的时长的计算公式如下:

$$\delta t = (pri - L2) * T$$

其中, $\delta t$ 为计数数值从所述初始值递减计数到第二预设数值所需的时长, $pri$ 为所述初始值, $L2$ 为所述第二预设阈值, $T$ 为时钟周期。

## 嵌入式系统及其计时方法

### 技术领域

[0001] 本发明涉及嵌入式系统技术领域,尤其涉及一种嵌入式系统及其计时方法。

### 背景技术

[0002] 嵌入式系统是半导体技术和软件技术面向应用的综合,已经有数十年的历史,如今已经随处可找到它们的踪影。各种消费电子产品、交通工具、通信产品、自动化控制系统等其中都会集成嵌入式系统,用于对整个产品、系统进行控制以及进行人机交互,从而让产品更加智能、灵活、易用,使产品具有更加强大的功能。

[0003] 在集成嵌入式系统的产品中,产品的很多功能是基于时间或定时的,如在特定的时间开/关机、特定时间执行某项功能等。在一些复杂、成本较高的嵌入式系统中,电路中会集成实时时钟电路,软件一般也会包含操作系统,实现较为准确的系统时间比较方便。但针对一些低成本的嵌入式系统,无论是所用微处理器,还是系统电路设计,出于成本等考虑,往往不集成实时时钟电路。另外,很多低成本的嵌入式系统甚至不使用操作系统,整个应用软件运行于单任务状态。在这些低成本的嵌入式系统中,要想获取比较精确的时间,并不是一件容易的事情。

### 发明内容

[0004] 有鉴于此,本发明提供一种嵌入式系统及其计时方法,能够在不增加实时时钟电路的基础上实现嵌入式系统的精确计时。

[0005] 为解决上述问题,本发明提供一种嵌入式系统,包括:

[0006] 时间校准模块,用于获取系统的校准时间;

[0007] 时钟电路,用于产生时钟信号;

[0008] 计时器,用于接收所述时钟信号,并根据接收到的所述时钟信号的数量进行计数,得到一计数数值,在所述计数数值达到预设数值时,生成一中断事件通知,并获取所述计数数值达到预设数值所需的时长;

[0009] 处理器,用于在接收到所述中断事件通知后,执行中断服务程序,所述执行中断服务程序包括:统计所述中断服务程序的执行次数,并根据所述时长、所述执行次数以及所述校准时间,计算系统当前时间。

[0010] 可选的,所述计时器还包括:

[0011] 第一计时模块,用于在每接收到一所述时钟信号时,将所述计数数值加一,并获取所述计数数值从所述计时器的初始值递增计数到第一预设数值所需的时长;或

[0012] 第二计时模块,用于在每接收到一所述时钟信号时,将所述计数数值减一,并获取所述计数数值从所述初始值递减计数到第二预设数值所需的时长;

[0013] 可选的,所述处理器还包括:

[0014] 计时器重置模块,用于在执行所述中断服务程序时,重置所述计时器的初始值。

[0015] 可选的,所述处理器还包括:

[0016] 误差获取模块,用于获取从接收到所述中断事件通知至重置所述计时器的初始值之间所述计时器的计数数值,作为误差值,并计算所述误差值与所述初始值的和值;

[0017] 其中,所述计时器重置模块,还用于将所述和值作为所述计时器的初始值,并重置所述计时器的初始值。

[0018] 本发明还提供一种嵌入式系统的计时方法,包括:

[0019] 接收时钟电路产生的时钟信号,并采用计时器根据接收到的所述时钟信号的数量进行计数,得到一计数数值,在所述计数数值达到预设数值时,生成一中断事件通知;

[0020] 获取所述计数数值达到预设数值所需的时长;

[0021] 获取系统的校准时间;

[0022] 在接收到所述中断事件通知后,执行中断服务程序,所述执行中断服务程序包括:统计所述中断服务程序的执行次数,并根据所述时长、所述执行次数以及所述校准时间,计算系统当前时间。

[0023] 可选的,所述根据接收到的所述时钟信号的数量进行计数,得到一计数数值的步骤之前还包括:

[0024] 为计数数值设置一初始值;

[0025] 所述根据接收到的所述时钟信号的数量进行计数,得到一计数数值的步骤包括:

[0026] 在每接收到一所述时钟信号时,将所述计数数值加一,并获取所述计数数值从所述初始值递增计数到第一预设数值所需的时长;或

[0027] 在每接收到一所述时钟信号时,将所述计数数值减一,并获取所述计数数值从所述初始值递减计数到第二预设数值所需的时长;

[0028] 可选的,所述计数数值从所述初始值递增计数到第一预设数值所需的时长的计算公式如下:

$$[0029] \quad \delta t = (L1 - pri) * T$$

[0030] 其中, $\delta t$ 为计数数值从所述初始值递增计数到第一预设数值所需的时长, $pri$ 为所述初始值, $L1$ 为所述第一预设阈值, $T$ 为时钟周期;

[0031] 所述计数数值从所述初始值递减计数到第二预设数值所需的时长的计算公式如下:

$$[0032] \quad \delta t = (pri - L2) * T$$

[0033] 其中, $\delta t$ 为计数数值从所述初始值递减计数到第二预设数值所需的时长, $pri$ 为所述初始值, $L2$ 为所述第二预设阈值, $T$ 为时钟周期。

[0034] 可选的,所述执行中断服务程序的步骤还包括:

[0035] 重置所述计时器的初始值。

[0036] 可选的,所述重置所述计时器的初始值的步骤包括:

[0037] 获取从接收到所述中断事件通知至重置所述计时器的初始值之间的所述计时器的计数数值,作为误差值,并计算所述误差值与所述初始值的和值;

[0038] 将所述和值作为所述计时器的初始值,并重置所述计时器的初始值。

[0039] 可选的,所述计数数值包括高位计数段和低位计数段,所述将所述和值作为所述计时器的初始值,并重置所述计时器的初始值的步骤包括:

[0040] 将所述和值作为所述计时器的初始值,仅重置所述计时器的初始值的高位计数段

的值。

[0041] 本发明具有以下有益效果：

[0042] 在实现精确计时的同时，不需要增加硬件设计的成本，不需辅助的电路设计，只需借助计时器通过软件的方式完成，不会引入任何的时间误差，并且可完全避免处理器执行程序指令用时所带来的时间累积误差。

### 附图说明

[0043] 图1为本发明实施例的嵌入式系统的一结构示意图；

[0044] 图2为本发明实施例的计数数值的结构示意图；

[0045] 图3为本发明实施例的嵌入式系统的计时方法的流程示意图；

[0046] 图4为本发明实施例的嵌入式系统的另一结构示意图；

[0047] 图5为本发明实施例的嵌入式系统的运行方法的流程示意图；

[0048] 图6为本发明实施例的嵌入式系统的执行中断服务程序的方法的流程示意图。

### 具体实施方式

[0049] 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。

[0050] 如图1所示为本发明实施例的嵌入式系统的一结构示意图，所述嵌入式系统包括：

[0051] 时间校准模块101，用于获取系统的校准时间；时钟电路102，用于产生时钟信号；

[0052] 计时器103，用于接收所述时钟电路产生的时钟信号，并根据接收到的所述时钟信号的数量进行计数，得到一计数数值，在所述计数数值达到预设数值时，生成一中断事件通知，并获取所述计数数值达到预设数值所需的时长；

[0053] 处理器104，用于在接收到所述中断通知事件后，执行中断服务程序，所述执行中断服务程序包括：统计所述中断执行程序的执行次数，并根据所述时长、所述执行次数以及所述系统的校准时间，计算系统当前时间。

[0054] 下面对上述各模块的结构和工作原理进行详细说明。

[0055] (1)时间校准模块

[0056] 本发明实施例中，可以在嵌入式系统运行后的任意时刻，通过时间校准模块101对系统时间进行校准，得到系统的校准时间。例如在系统初始化后进行，或者在计时器初始化后进行。

[0057] 此外，所述时间校准模块101可以通过多种方式对系统时间进行校准，例如，通过提供一人机交互界面由用户输入时间，或者，通过连接其他系统通过私有或公用协议完成系统时间的同步和校准，例如通过NTP(Network Time Protocol,网络时间协议)协议对系统时间进行校准。

[0058] (2)时钟电路

[0059] 时钟电路102中可以包括一时钟源和一震荡电路，所述时钟源可以是一石英晶体或其他替代元件，所述时钟源和所述震荡电路相互配合，产生时钟信号。

[0060] 此外，所述时钟电路102除了具有产生时钟信号的功能之外，还可以具有对时钟信号进行分频、倍频等的功能，即将时钟源产生的时钟信号转换成多个不同频率的时钟信号，并分别发送给所述嵌入式系统中的需要不同频率的时钟信号的多个电路模块。

[0061] (3) 计时器

[0062] 本发明实施例中,可以首先为所述计时器103的计数数值设置一初始值pri,将所述计时器103设置成在所述初始值pri的基础上递增计数,或者,将所述计时器103设置成在所述初始值pri的基础上递减计数。

[0063] 下面分别对所述计时器103递增计数和递减计数的过程进行说明。

[0064] 当所述计时器103被设置成递增计数时,所述计时器103每接收到所述时钟电路102发送过来的一个时钟信号时,便将计数数值加一,直至所述计时器103的计数数值达到第一预设数值L1(例如计时器103对应的溢出值),当所述计时器103的计数数值达到第一预设数值L1时,生成一中断事件通知,通知所述处理器104执行中断服务程序。

[0065] 此外,所述计时器103还需要计算所述计数数值从所述初始值递增计数到第一预设数值所需的时长 $\delta t$ ,所述时长 $\delta t$ 的计算公式如下:

$$[0066] \quad \delta t = (L1 - pri) * T$$

[0067] 其中, $\delta t$ 为计数数值从所述初始值递增计数到第一预设数值所需的时长,pri为所述初始值,L1为所述第一预设阈值,T为时钟周期。

[0068] 当所述计时器103被设置成递减计数时,所述计时器103每接收到所述时钟电路102发送过来的一个时钟信号时,便将计数数值减一,直至所述计时器103的计数数值达到第二预设数值L2(例如零值),当所述计时器103的计数数值达到第二预设数值L2时,生成一中断事件通知,通知所述处理器104执行中断服务程序。

[0069] 此外,所述计时器103还需要计算所述计数数值从所述初始值递减计数到第二预设数值所需的时长 $\delta t$ ,所述时长 $\delta t$ 的计算公式如下:

$$[0070] \quad \delta t = (pri - L2) * T$$

[0071] 其中, $\delta t$ 为计数数值从所述初始值递减计数到第二预设数值所需的时长,pri为所述初始值,L2为所述第二预设阈值,T为时钟周期。

[0072] 基于上述描述,本发明实施例的计时器103还可以包括:

[0073] 第一计时模块,用于在每接收到一所述时钟信号时,将所述计数数值加一,并获取所述计数数值从所述计时器的初始值递增计数到第一预设数值所需的时长;或

[0074] 第二计时模块,用于在每接收到一所述时钟信号时,将所述计数数值减一,并获取所述计数数值从所述初始值递减计数到第二预设数值所需的时长。

[0075] (4) 处理器

[0076] 处理器104在接收到计时器103发送的中断事件通知时,执行一中断服务程序,执行中断服务程序包括:统计所述中断服务程序的执行次数CNT,并根据所述执行次数CNT、所述计时器103计算得到的时长 $\delta t$ 以及所述系统的校准时间,计算系统当前时间。

[0077] 所谓中断,是嵌入式系统为计时器103提供的一种机制,当计时器103的计数数值达到预设数值(例如计数数值溢出时)时,则产生一中断事件通知,并第一时间发送给处理器104。

[0078] 本实施例中,可以预先为计时器103的中断注册一个中断服务程序,当计时器103产生中断时,处理器104会临时停止当前执行的任务(本实施例中不考虑中断优先级、任务调度等问题),执行计时器103对应的中断服务程序,完成后再返回去执行先前的任务。

[0079] 下面对中断服务程序执行时,计算系统当前时间的方法进行详细说明,主要包括

以下步骤:

[0080] 1)获取系统的校准时间。

[0081] 在系统初始化后,可以首先进行系统时间的校准,从外部获取准确的时间(例如通过NTP协议获取时间或者提供人机交互界面由用户直接输入时间),将从外部获取的时间作为校准时间。此外,在进行系统时间校准时,还需要同时将中断服务程序的执行次数CNT清零。

[0082] 举例来说,校准时间为2011年10月10日3点3份3秒。

[0083] 2)获取计时器的计数数值达到预设数值所需的时长 $\delta t$ 。

[0084] 3)统计中断服务程序的执行次数CNT。

[0085] 4)计算CNT与 $\delta t$ 的乘积。

[0086] CNT\* $\delta t$ 的值代表了计时器103的工作时长。

[0087] 5)计算校准时间与CNT\* $\delta t$ 之和,得到系统当前时间。

[0088] 假如计算出的CNT\* $\delta t$ 的值为864008秒(10天零8秒),系统当前时间即为2011年10月20日3点3分11秒。

[0089] 本发明的其他实施例中,所述处理器104也可以通过其他计算方法计算系统当前时间。

[0090] 举例来说,计时器103会设置一个系统初始时间,例如为1970年1月1体0点0分0秒,处理器104可以采用获取到的校准时间减去系统初始时间,得到一个时间差值M,并将CNT的值修改为CNT+(M/ $\delta t$ ),然后计算系统初始时间与CNT\* $\delta t$ 之和,得到系统当前时间。

[0091] 本发明实施例中,所述处理器104还可以在每次执行中断服务程序时,将计时器103的计数数值重置成初始值pri。在处理器104执行其他任务,包括校准系统时间等,计时器103仍会按照自身的既定节拍运行,并在计时器到达计时阈值时产生计时器中断。

[0092] 但是,每次执行中断服务程序的时候,简单将计数器103的值重置为初始值pri会产生一个时间误差,下面对这个时间误差的产生原因和消除该误差的方法做说明。

[0093] 上述实施例中,处理器104在接收到中断事件通知后,需要先处理完当前正在处理的任任务片段,并保护现场,然后再执行中断服务程序,本发明实施例中,将处理器104在接收到中断事件通知到开始执行中断服务程序之间的这段时间称为中断潜伏期。

[0094] 由于中断潜伏期的存在,处理器104在重置计时器的初始值时,如果简单将计时器103的初始值重置为pri,会出现计时误差,具体理由如下:

[0095] 假设从计时器产生中断到计时器中断服务程序被执行的时间长度为 $t_1$ ,处理器从开始执行计时器中断服务程序到重置计时器的计数数值的时间长度为 $t_2$ ,计时器被重置为pri到计时器溢出的时间为 $\delta t$ 。

[0096] 那么,可以看出如果计时器被简单重置为初始值pri,则计时器每溢出一次(计时器中断服务程序被执行一次)所用的时间应该为 $\delta t+t_1+t_2$ 。其中, $t_1$ 和 $t_2$ 就是累计时间时产生的误差,这两个值一般很小且值不固定,但长期累积就会产生巨大的误差,需要将其消除掉,将计时器中断服务程序被执行的平均间隔时间精确控制在 $\delta t$ 值上。

[0097] 下面对如何消除上述由软件引入的计时误差的方法进行详细说明。

[0098] 本发明实施例中,可以通过调整重置的计时器103的计数数值,将“中断潜伏期”+“中断服务程序执行到重置计数数值所需时间”这个时间误差减掉。



[0099] 具体消除过程如下：

[0100] 当计时器103以递增方式计数时，在计时器103溢出后，可以将计时器103设置为将计数数值自动清零，并继续从0开始递增计数。假设中断服务程序重置计时器的初始值时，计时器的计数数值为 $y$ ，则 $y*T$  = “中断潜伏期” + “中断服务程序执行到重置计数数值所需时间”。此时，可以将 $pri+y$ 作为新的初始值，重置计时器的计数数值。

[0101] 如果直接将计数数值重置为 $pri+y$ ，则可能也会引入误差，因为处理器在执行重置操作时（该重置操作包括读取 $y$ 值，并与 $pri$ 相加，然后将得到相加和写回计时器），需要执行时间，而在此期间，计时器仍在不断递增，从而引入、增加了“ $y$ 与 $pri$ 相加、相加和写入计时器等”指令执行所需要时间。

[0102] 为了解决重置操作引起的计时误差，本发明实施例中，如图2所示，可以将计数数值分为两段高位计数段(Hreg)和低位计数段(Lreg)，其中，低位计数段从 $bit[0] \sim bit[x+1]$ ，高位计数段从 $bit[x+2] \sim bit[n]$ ，高位计数段和低位计数段共具有 $n+1$ 个二进制比特。

[0103] 另外，需要说明的是，本发明实施例中，设置初始值 $pri$ 时，需要确保 $pri$ 的值转换为二进制时对应的 $bit[0] \sim bit[x+1]$ 的值都为0，确保 $\delta t$ 值小于1秒，以便获得一个合适的最小时间粒度。最好使的1秒恰好是 $\delta t$ 的整数倍，如 $\delta t$ 取值50毫秒、100毫秒、500毫秒等值，以便在执行中断服务程序时方便地根据 $\delta t$ 和CNT值（中断服务程序被执行的次数）计算年月日时分秒的值； $\delta t$ 不要太小，否则中断服务程序会被过度频繁执行，消耗大量的CPU计算运算资源。

[0104] 假设所述计时器103被设置成递增计数，所述计时器103每接收到所述时钟电路102发送过来的一个时钟信号，便将计数数值加一，直至所述计时器103的计数数值溢出（计数数值达到 $2^{n+1}$ ， $2^{n+1}$ 即上述预设数值），当所述计时器103的计数数值溢出时，生成一中断事件通知，通知所述处理器104执行中断服务程序。

[0105] 此外，所述计时器103还需要计算所述计数数值从所述初始值递增计数到溢出值所需的时长 $\delta t$ ， $\delta t$ 的计算公式如下：

$$[0106] \quad \delta t = (2^{n+1} - pri) * T$$

[0107] 其中， $pri$ 为计数数值的初始值， $2^{n+1}$ 为计数数值的溢出值， $T$ 为时钟周期，\*号为乘号。

[0108] 假设计数数值为 $pri$ 时，Hreg的值为 $P$ ，Lreg的值为 $Q$ （由上文提到的 $pri$ 取值原则可知， $Q=0$ ）；计数数值为 $y$ 时，Hreg的值为 $p$ ，Lreg的值为 $q$ 。

[0109] 由此可得出：

$$[0110] \quad pri+y = P*(2^{x+2}) + Q + p*(2^{x+2}) + q = (P+p)*(2^{x+2}) + q$$

[0111] 本发明实施例中，重置计数数值时，只需对Hreg进行重置，而Lreg继续进行递增计数，Lreg继续计数的目的就是要消除 $t1+t2$ 这个误差。

[0112] 上面提到了，计时器溢出时，计数数值会清零，并从0开始递增计数。当处理器经过了中断潜伏期（时间长度为 $t1$ ），开始执行计时器中断服务程序，从开始执行中断服务程序到执行重置计时器（Hreg, Lreg）用时为 $t2$ ，在这个过程中，Lreg还在继续计数，且其起始值为0。当在中断服务程序中执行重置计时器指令的那一刻，只需要查看Lreg此时的计数值就能够知道想知道 $t1+t2$ 的值了，假定这一时刻的Lreg值为 $q$ ，那么 $t1+t2 = q*T$ 。其中， $T$ 为计时器输入时钟信号的周期。

[0113] 重置计数数值的初始值时,不是简单重置为pri,而是重置为pri+q,那么下次计时器溢出的时间间隔就不是 $\delta t$ 时长了,而是 $\delta t-(t_1+t_2)$ ,这样就会把上次中断产生时引入的 $t_1+t_2$ 这个时间误差给抵消掉。

[0114] 本发明实施例中,将处理器执行计数数值重置指令所需要的时间也考虑进来,在重置计数数值时,只对Hreg部分进行操作,Lreg不操作(继续做计数),处理器执行计数数值重置指令所需要的时间也会被抵消掉,以便保证计时程序完全不会引入计时时间误差。

[0115] 另外,在对计数数值进行Hreg、Lreg分段的时候,Lreg段共有x+2个bit位,其最大值为qmax,其中, $qmax*T >$ “中断潜伏期”+“中断服务程序执行到重置计数数值所需时间”,也就是在中断事件发生到重置计数数值的过程中,Lreg位段不能向Hreg位段进位。

[0116] 上述实施例中,是以计时器递增计数为例进行说明。

[0117] 当计时器103被设置成递减计数,所述计时器103每接收到所述时钟电路102发送过来的一个时钟信号,便将计数数值减一,直至所述计时器103的计数数值归零(此时,零值即上述预设数值),当所述计时器103的计数数值归零时,生成一中断事件通知,通知所述处理器104执行中断服务程序。

[0118] 此外,所述计时器103还需要计算所述计数数值从所述初始值递减计数到零值所需的时长 $\delta t$ , $\delta t$ 的计算公式如下:

[0119]  $\delta t = pri * T$

[0120] 其中,pri为计数数值的初始值,T为时钟周期,\*号为乘号。另外,为了消除计时误差,所述计时器103的计数数值归零后,可以将计时器103的计数数值设置为 $2^{n+1}-1$ ,并继续递减计数,以获取计时误差。

[0121] 基于上述描述,本发明实施例的处理器还包括:

[0122] 误差计算模块,用于获取从接收到所述中断事件通知至重置所述计时器的初始值之间所述计时器的计数数值;

[0123] 所述计时器重置模块,还用于在执行所述中断服务程序时,将所述计时器的初始值重置为预设初始值加所述中间潜伏期及部分程序指令执行时间对应的计数数值。

[0124] 通过上述实施例提供的嵌入式系统,在实现精确计时的同时,不需要增加硬件设计的成本,不需辅助的电路设计。只需借助计时器通过软件的方式完成,不会引入任何的时间误差。嵌入式系统的时间精度仅与时钟源的精度相关,经过实验和理论计算得知,采用50ppm精度的时钟源,本发明实施例的嵌入式系统每天可能出现的最大时间误差是4.32秒,采用10ppm精度的时钟源,本发明实施例的嵌入式系统每天可能出现的最大时间误差是0.864秒。

[0125] 对应于上述嵌入式系统,本发明实施例还提供一种嵌入式系统的计时方法,如图3所示,所述嵌入式系统的计时方法包括以下步骤:

[0126] 步骤301,接收时钟电路产生的时钟信号,并采用计时器根据接收到的所述时钟信号的数量进行计数,得到一计数数值,在所述计数数值达到预设数值时,生成一中断事件通知;

[0127] 步骤302,获取所述计数数值达到预设数值所需的时长;

[0128] 步骤303,获取系统的校准时间;

[0129] 步骤304,在接收到所述中断事件通知后,执行中断服务程序,所述执行中断服务

程序包括:统计所述中断服务程序的执行次数,并根据所述时长、所述执行次数以及所述校准时间,计算系统当前时间。

[0130] 本发明实施例中,可以首先为计数数值设置一初始值,将所述计时器设置成在所述初始值的基础上递增计数,或者,将所述计时器设置成在所述初始值的基础上递减计数。

[0131] 即,上述根据接收到的所述时钟信号的数量进行计数,得到一计数数值的步骤包括:

[0132] 在每接收到一所述时钟信号时,将所述计数数值加一,并获取所述计数数值从所述初始值递增计数到第一预设数值所需的时长;或

[0133] 在每接收到一所述时钟信号时,将所述计数数值减一,并获取所述计数数值从所述初始值递减计数到第二预设数值所需的时长;

[0134] 所述计数数值从所述初始值递增计数到第一预设数值所需的时长的计算公式如下:

$$[0135] \quad \delta t = (L1 - pri) * T$$

[0136] 其中, $\delta t$ 为计数数值从所述初始值递增计数到第一预设数值所需的时长, $pri$ 为所述初始值, $L1$ 为所述第一预设阈值, $T$ 为时钟周期;

[0137] 所述计数数值从所述初始值递减计数到第二预设数值所需的时长的计算公式如下:

$$[0138] \quad \delta t = (pri - L2) * T$$

[0139] 其中, $\delta t$ 为计数数值从所述初始值递减计数到第二预设数值所需的时长, $pri$ 为所述初始值, $L2$ 为所述第二预设阈值, $T$ 为时钟周期。

[0140] 为了消除软件引入的计时误差,本发明实施例中,所述执行中断服务程序的步骤还包括:重置所述计时器的初始值。

[0141] 所述重置所述计时器的初始值的步骤包括:

[0142] 获取从接收到所述中断事件通知至重置所述计时器的初始值之间的所述计时器的计数数值,作为误差值,并计算所述误差值与所述初始值的和值;

[0143] 将所述和值作为所述计时器的初始值,并重置所述计时器的初始值。

[0144] 为了消除软件计时误差,所述计数数值被分为高位计数段和低位计数段,所述将所述和值作为所述计时器的初始值,并重置所述计时器的初始值的步骤包括:将所述和值作为所述计时器的初始值,仅重置所述计时器的初始值的高位计数段的值。

[0145] 通过上述实施例提供的计时方法,在实现精确计时的同时,不需要增加硬件设计的成本,不需辅助的电路设计。只需借助计时器通过软件的方式完成,不会引入任何的软件执行所带来的时间误差。嵌入式系统的时间精度仅与时钟源的精度相关。

[0146] 如图4所示为本发明实施例的嵌入式系统的另一结构示意图,所述嵌入式系统包括:微处理器、时钟源、RAM和ROM,其中微处理器又包括:CPU核心单元(相当于上述实施例中的处理器104)、计时器(相当于上述实施例中的计时器103)和时钟电路。其中,时钟源和所述时钟电路配合相当于上述实施例中的时钟电路102。

[0147] 如图5所示,本实施例的嵌入式系统的运行过程如下:

[0148] 步骤501,系统上电启动;

[0149] 步骤502,系统初始化;

- [0150] 步骤503, 计时器初始化, 注册中断服务程序;
- [0151] 其中, 计时器初始化包括: 设置计时器的初始值; 设置计时器递增或递减计数; 输入时钟周期T等。
- [0152] 步骤504, 校准系统时间;
- [0153] 步骤505, 运行其他各项任务。
- [0154] 如图6所示, 本实施例的时嵌入式系统执行中断服务程序的过程包括以下步骤:
- [0155] 步骤601, 进入中断服务程序;
- [0156] 步骤602, 重置计时器的计数数值的值;
- [0157] 步骤603, 统计中断服务程序的执行次数CNT;
- [0158] 步骤604, 根据 $\delta t$ 和CNT, 计算系统当前时间;
- [0159] 步骤605, 退出中断服务程序。
- [0160] 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, 在不脱离本发明原理的前提下, 还可以作出若干改进和润饰, 这些改进和润饰也应视为本发明的保护范围。

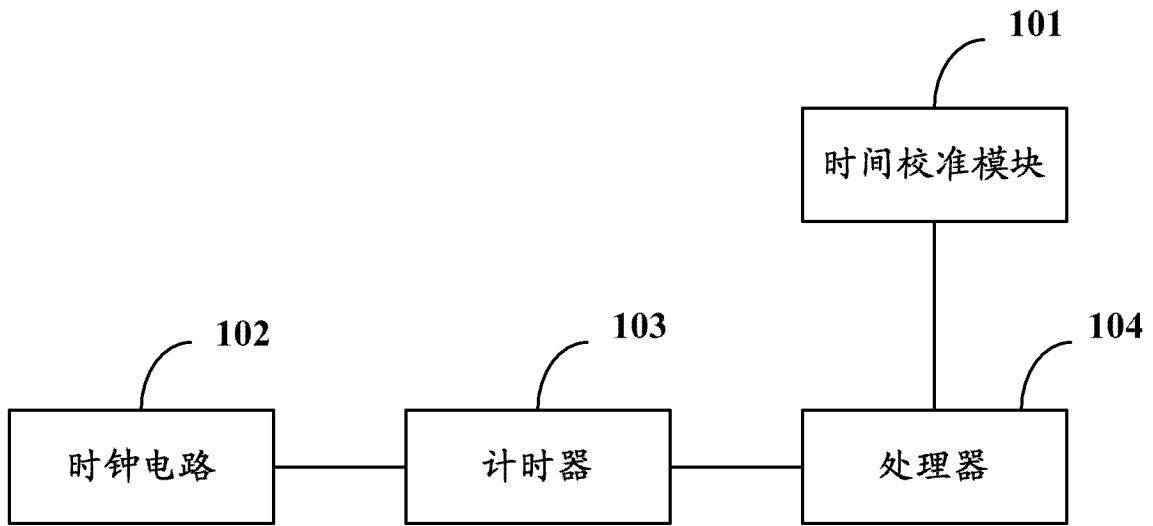


图1

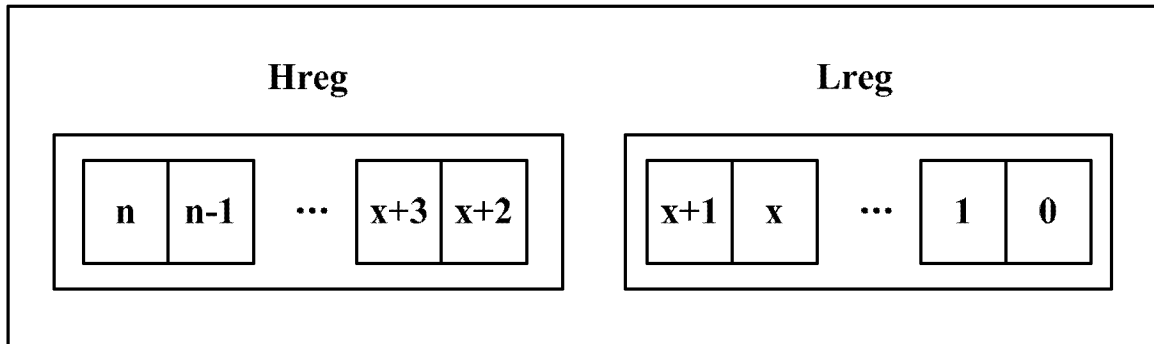


图2

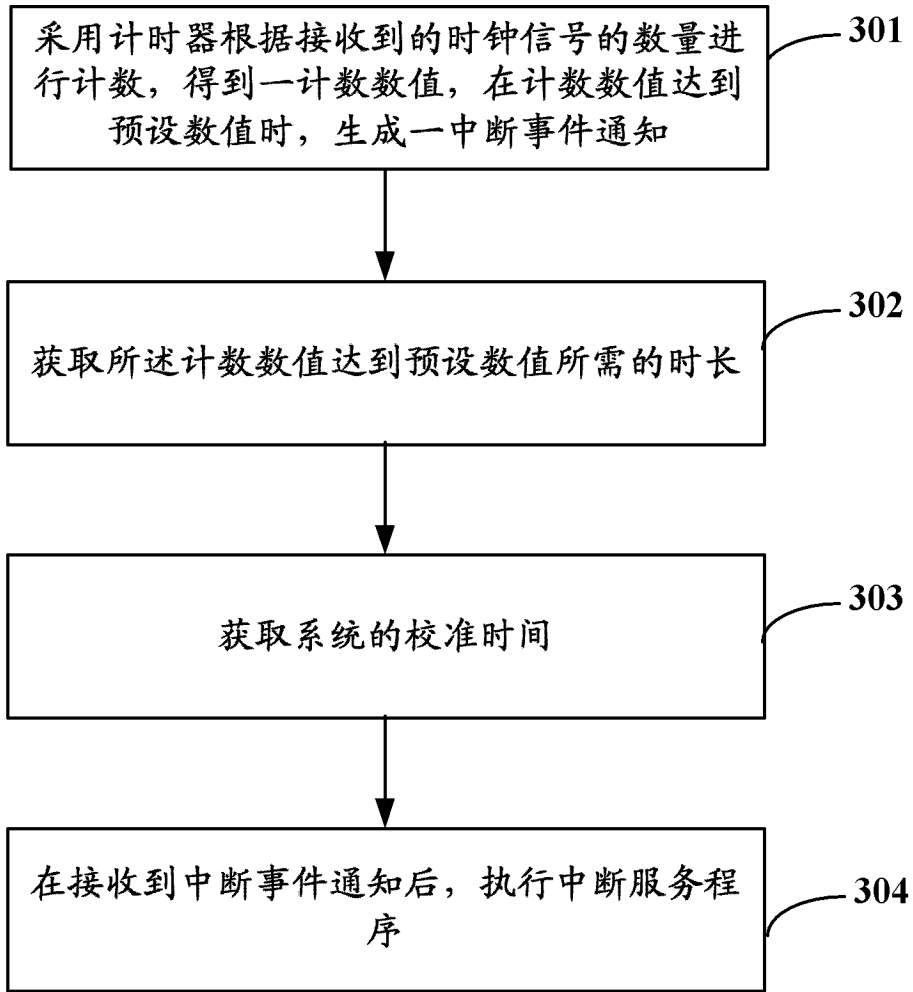


图3

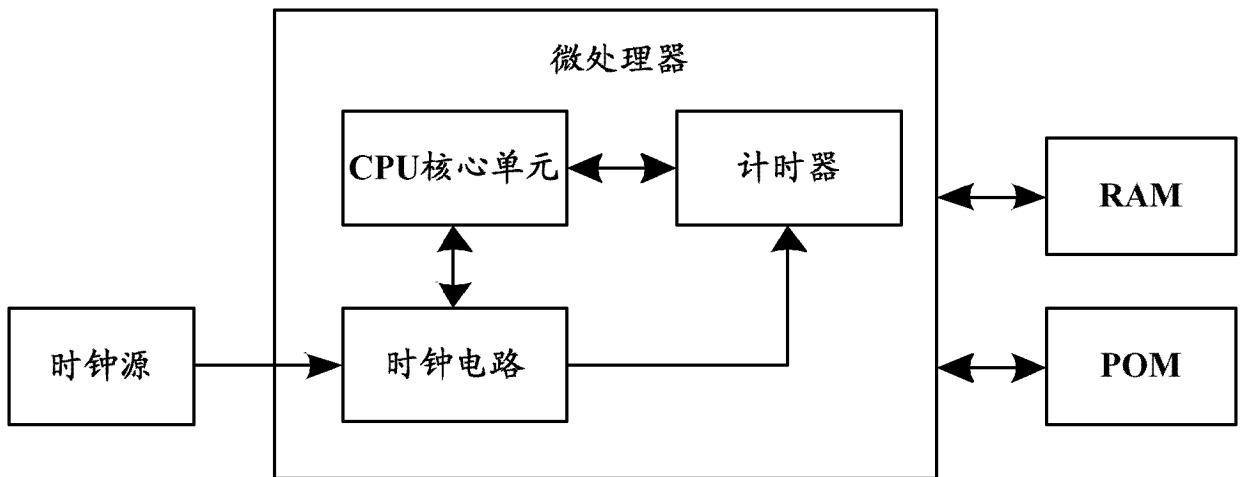


图4

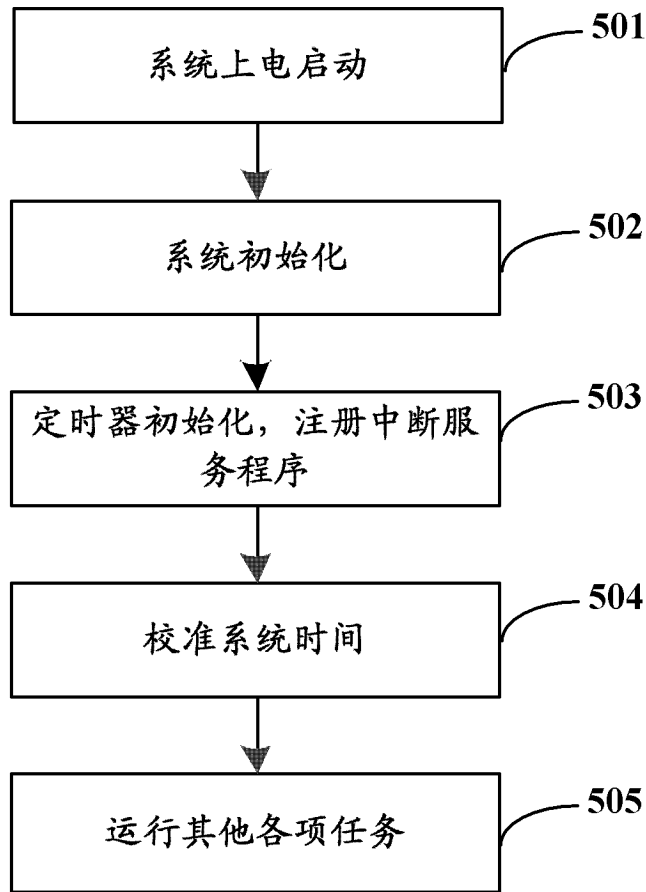


图5

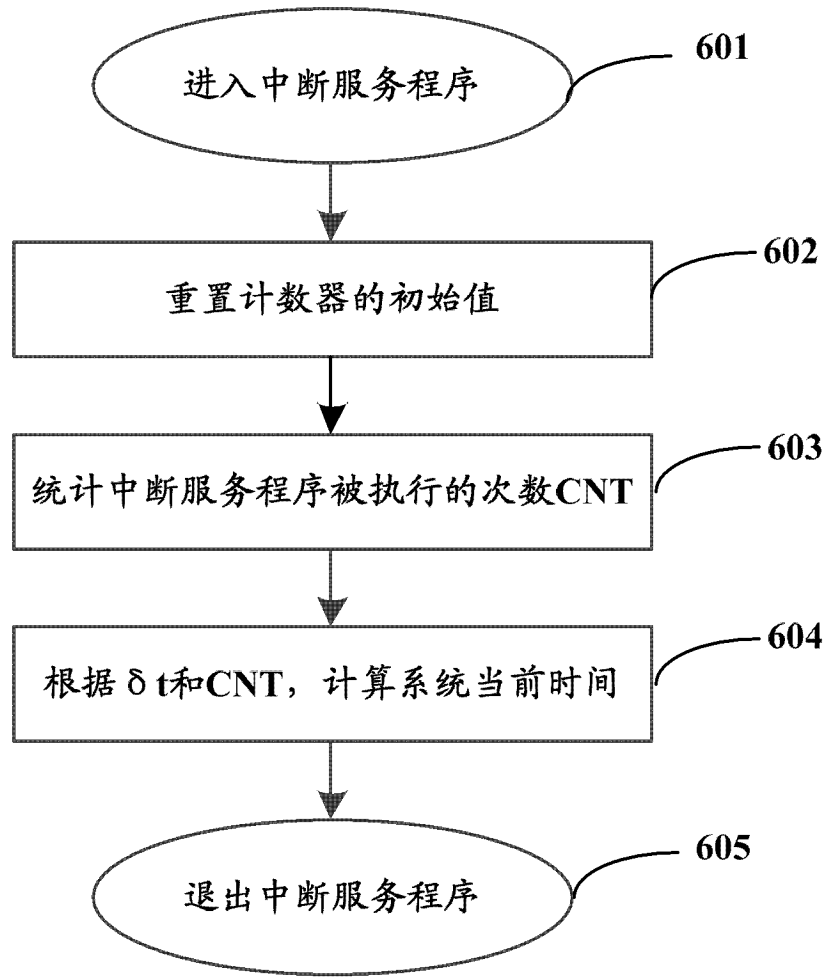


图6