

基于 1553B 总线的卫星星时高精度自主恢复设计

鲁超¹, 杨丽君¹, 陈伯翰², 张睿¹, 李林¹

(1. 北京空间飞行器总体设计部, 北京 100094;

2. 中国空间技术研究院 载人航天总体部, 北京 100094)

摘要: 随着现代航天器功能日趋复杂, 在轨任务不断变化, 星上各个系统对于时间的依赖性也随之不断提高, 为此也对星上时间管理的可靠性、安全性提出了较高的要求; 为了实现地球同步轨道卫星和深空探测卫星等无法携带 GPS 接收机的航天器在异常情况下的星时自主恢复, 文章提出了一种基于 1553B 总线的星时高精度自主恢复方案, 利用制定的软件协议, 在不额外增加硬件开销的前提下, 实现了星时自主恢复; 经过实验验证方案可行, 实现了高精度的星时自主恢复; 此方法能够有效减少对地面人工干预的依赖, 并能够有效地保证在意外情况下星载数据管理计算机工作的连续性, 大大提高了卫星时间管理系统在星载数据管理计算机出现复位或切机等异常情况下的可用性, 在实际工程中具有一定的实用价值。

关键词: 卫星; 星上时间; 自主恢复

Satellite Time Precise Auto-Recovery Design Based on MIL-STD-1553 Bus

Lu Chao¹, Yang Lijun¹, Chen Bohan², Zhang Rui¹, Li Lin¹

(1. Beijing Institute of Spacecraft System Engineering, Beijing 100094, China;

2. Institute of Manned Spacecraft System Engineering, China Academy of Space Technology, Beijing 100094, China)

Abstract: With the increasing complexity and the variety of spacecraft functions and tasks, the dependence on satellite time has been improved for various on-board systems of spacecraft. Therefore, the higher reliability and safety requirements have been brought to the satellite time management function. A satellite time precise auto-recovery design based on MIL-STD-1553 bus has been brought forward for the spacecraft that cannot carry the GPS receiver, such as GEO satellite and space exploration satellite. The design uses the developed software protocol to realize the satellite time precise auto-recovery without additional hardware overhead. The experiment proves that this design is feasible. It can effectively reduce manual intervention, and can ensure the continuity of the onboard computer, and greatly improve the availability of the time system. It has a certain practical value in engineering.

Keywords: satellite; satellite time; onboard-time; automatic recovery

0 引言

卫星星时管理和维护是航天器数据管理的一项基本任务, 通常是由星载数据管理计算机软件实现的。卫星星时可用于支持下行遥测信息的形成, 支持延时命令的执行, 支持姿态和轨道控制以及其他分系统的运行, 也支持各类遥感图像及收集到的电子信号的精确定位等^[1], 因此卫星星上时间对于卫星的正常运行和应用起到了至关重要的作用。由于卫星工作的特殊空间环境, 星载数据管理计算机受到空间环境变化及空间天气灾害事件的影响, 造成星载数据管理计算机自主复位、切换备份等操作, 影响星地时间同步, 进而影响与星时相关任务的执行。因此, 引入星时自主恢复机制对于卫星数据管理是十分必要的。

对于携带 GPS 接收机的航天器, 可以通过接收 GPS 时

间达到自主恢复星时的目的, 但是对于地球同步轨道卫星和深空探测卫星等没有携带 GPS 接收机的航天器而言, 这种方法不适用。另外, 美国已成功掌握了局部屏蔽 GPS 信号等技术^[2], 这就对依赖于 GPS 的国内航天器带来了潜在的应用风险^[3]。本文针对没有携带 GPS 接收机的航天器, 星时精度要求高的卫星系统提出了一种基于 1553B 总线的星时高精度自主恢复设计。

1 星时管理方案简介

典型的卫星系统中, 星时管理是由星载数据管理计算机实现的, 下面对目前航天器星时管理的相关概念进行介绍:

星时生成: 由卫星系统和地面测控系统约定一个时间点作为时间零点 (例如北京时间 2008 年 1 月 1 日 0: 00: 00 作为时间零点), 然后由星载数据管理计算机计时电路计时, 产生的相对时间, 作为卫星星时。星时一般为相对于其时间零点的累积值, 卫星发射前需根据时间零点对累计

收稿日期: 2018-04-23; 修回日期: 2018-07-01。

作者简介: 鲁超(1976-), 男, 高级工程师, 主要从事卫星数据管理系统方向的研究。

值进行设置, 一般在星载数据管理计算机中由秒部分和亚秒部分组成, 秒部分计数单位为 1 秒, 亚秒部分计数单位为若干毫秒或若干微秒 (由星时计时电路最小分辨率和亚秒部分数据长度确定), 其结构形式如表 1 所示^[4]。

表 1 星上时间码结构表

$b_0 \dots b_i$	$b_{i+1} \dots b_n$
时间码(秒)	时间码(亚秒)

星时分发: 常用的有两种方式, 一种由星载数据管理计算机每秒送出秒脉冲给星时用户使用, 用户收到秒脉冲后当作自己的秒计数用; 另一种是由星载数据管理计算机以固定时间间隔在总线上广播或者发送当前的星时, 星时用户接收时间码, 扣除误差后作为自己的星时。在卫星系统里, 除星载数据管理计算机外的星时用户, 通常有载荷分系统和控制分系统。

星时下载: 由星载数据管理计算机以固定的间隔取出星时填入卫星遥测帧, 通过遥测帧下传至地面。地面测控系统接收到遥测数据中的星时后, 根据约定的时间零点以及星时数据, 计算出当前星时。

星时维护: 大多数航天器软件采用通过秒中断计数来维护星上时间的方法, 维护由计数器产生的时间信息^[5-6]。

星时校准: 对星上时间校准采用了集中校时和均匀校时两种方法。在这两种方法中, 飞行器定时将星上时间传回地面进行比对, 地面计算出星上与地面时钟的时间误差, 集中校时就是把地面测定的时间误差 T_0 通过遥控注入星上, 星上把接收到的时间差一次性加到时间计数器中, 从而纠正星上时钟; 均匀校时是地面根据计算出的时间误差拟合曲线 ($kt + T_0$), 然后将获得的参数 k 通过遥控注入星上, 星上时钟就按照每秒加 k 进行修正。集中校时可以消除累计误差, 均匀校时有利于保持星上时间的连续性, 在使用中通常两种方法配合进行^[7]。

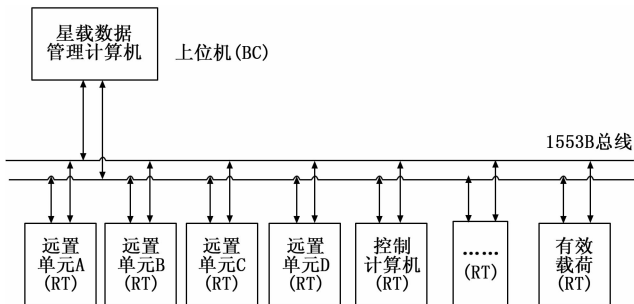


图 1 典型的数据管理分系统总线拓扑图^[8]

2 星时恢复方案

常规的星时恢复有两种, 一种是通过地面测量星地时差, 当时差累计到某一个设定的阈值时, 生成校时数据对星时进行调整, 这种方法需要地面人工干预, 同时不能保证实时性, 只能是事后处理, 这种方法对于卫星不可见弧

段的工作会有很大影响, 尤其是太阳同步轨道的遥感卫星, 其主要大部分工作均在不可见弧段进行; 另一种是将星时作为重要数据, 定期保存, 当星载数据管理计算机出现自主复位、切换备份情况时, 自动从保存的重要数据中取回星时, 此种方法不能保证恢复后的星时误差, 误差取决于重要数据保存间隔和计算机自主复位或切换备份所需要时间, 一般误差在十秒以上, 不能满足对星时精度要求高的卫星系统的要求, 其过程如图 2 所示。

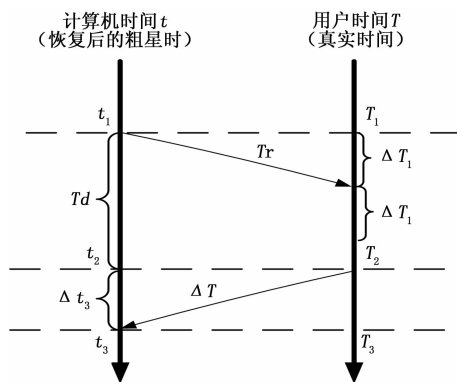


图 2 星时恢复过程示意图

基于 1553B 总线的卫星星时高精度自主恢复设计是基于 1553B 总线通信协议, 在星载数据管理计算机复位、切换备份时, 自主通过总线从星时用户取回相对星时时差, 并对自身时间码进行自动校准。图 1 为典型的数据管理分系统总线拓扑图, 在该系统内计算机可将卫星星时定时通过 1553B 总线分发到星上用户, 并在计算机复位或者切机时, 通过预先制定的通信协议从星上时间用户取回时间码进行自动校准, 从而完成星时恢复。具体星时恢复设计如下:

1) 选择星时恢复来源:

选定一个星时用户作为星时自主恢复的来源。该用户使用计算机分发的时间作为本机时间, 并且星时用户在两次时间分发间隔内具有较精确的星时维护能力。

时间用户则是时间维护和发布时间发布的对象, 控制分系统、载荷、数传分系统及地面测控均可认为是时间用户^[9-10], 目前航天器上主要选择控制计算机或载荷计算机作为星时恢复来源, 也可以同时选择两个星时用户作为星时自主恢复的来源, 事先设置恢复的优先顺序, 并设置合理的判据。

2) 重要数据保存和恢复:

在传统数据管理系统, 星载数据管理计算机会将软件运行中产生的重要的数据通过总线定时 (一般为 1 分钟) 发送给远置单元进行保存, 如果计算机复位或切换, 软件启动后, 首先从保存的重要数据进行星时恢复, 作为当前粗略的星时, 该星时 t 落后于当前真实时间 T 。

重要数据保存一般选择长期加电工作的 RT 终端, 为了保证星上重要数据的安全, 可以选择同时保存在两个 RT 终

端中, 事先设置恢复的优先顺序, 第一个恢复不成功则继续从第二个恢复, 这样可以保证其中一个 RT 异常时仍然能够顺利恢复重要数据。

3) 发送从重要数据恢复的粗略星时:

星载数据管理计算机取出当前恢复后的星时 t_1 (此时星时用户时间为 T_1), 通过 1553B 总线 BC-RT 方式发送给选定的星时用户, 并将取出星时到星时通过 1553B 总线送到星时用户总线控制器缓存中这个过程中产生的延时修正量 Δt_1 修正到发送的星时数据中。因此, 星时用户接收到的时间数据 Tr 为:

$$Tr = t_1 + \Delta t_1$$

星时用户接收到 1553B 总线发送的时间数据时, 实际经历时间为 ΔT_1 。

修正量 Δt_1 可以通过软件仿真分析、调试、测试得到, 需要事先进行标定, 使得 Δt_1 的值接近 ΔT_1 , 标定结果越准确, 则产生的误差越小。

记 $\Delta_1 = \Delta T_1 - \Delta t_1$, 为修正量 Δt_1 与实际经历时间 ΔT_1 之间的误差。

4) 星时用户计算时差:

选定的星时用户的 1553B 总线控制器会产生中断信号, 该中断需要星时用户的软件以高优先级的方式进行处理, 因此星时用户需及时将缓存中的星时数据读出, 然后计算当前时刻与接收到星时 Tr 的差, 同时要考虑从缓存中读出到计算开始这个过程的延时修正量 Δt_2 , 从收到时间数据到计算开始实际经历的时间为 ΔT_2 , 因此计算后的时差 ΔT 为:

$$\Delta T = T_1 + \Delta T_1 + \Delta T_2 - Tr - \Delta t_2 = T_1 - t_1 + \Delta T_1 - \Delta t_1 + \Delta T_2 - \Delta t_2$$

记 $\Delta_2 = \Delta T_2 - \Delta t_2$, 为修正量 Δt_2 与实际经历时间 ΔT_2 之间的误差。

此时:

$$\Delta T = T_1 - t_1 + \Delta_1 + \Delta_2$$

同样, Δt_2 也需要通过软件仿真分析、调试、测试进行事先标定, 标定结果越准确, Δt_2 的值越接近 ΔT_2 , 产生的误差也越小。

5) 星时用户发送时差数据:

星时用户计算出时差数据后有两种方式将此数据送给星载数据管理计算机: 一是将时差数据放入约定地址后, 通过 1553B 总线向星载数据管理计算机提出服务请求; 二是按照约定的时间内, 将时差数据放入约定地址, 等待星载数据管理计算机自动取走。前者实时性好, 能够缩短星时恢复过程所需时间。

6) 计算机修正星时:

数据管理计算机在等待一定时间 (T_d) 后开始从星时用户那里, 通过 1553B 总线 RT-BC 方式将时间差读回 (假设延时为 Δt_3), 并对读回的数据有效性进行判断, 当数据有效时对当前星时 t_3 进行修正, t_3 为:

$$t_3 = t_1 + T_d + \Delta t_3$$

因此修正后的时间 t 为:

$$t = t_3 + \Delta T + \Delta_3 = T_1 + T_d + \Delta t_3 + \Delta_1 + \Delta_2$$

如果读回的数据无效, 并且实现设置了两个星时用户用于星时恢复, 则按照约定向第二个星时用户重复 (3) ~ (6) 操作。

3 实验结果与分析

根据上述星时恢复方案, 在地面利用星载数据管理计算机和一台载荷计算机进行了测试, 两台计算机均使用外部同一个时钟源进行时间计数, 星载数据管理计算机时间亚秒计数单位为 $16 \mu s$ 。设计了如下测试方案及步骤:

第一步, 建立时间校准系统。使用 GPS 作为地面时间基准, 测量下行遥测信号星地时延, 再采用下传的遥测帧中的时间码与地面时间基准比较的方法, 建立时间校准系统, 具备均匀校时、集中校时功能。

第二步, 校准卫星时间。使用时间校准系统, 对星载数据管理计算机进行校时, 使之与 GPS 时间同步, 等待星载数据管理计算机分发星时, 载荷计算机利用收到的星时完成自身校时。

第三步, 复位星载数据管理计算机并恢复星时。地面发送指令使星载数据管理计算机复位, 等待复位完成并从载荷计算机恢复星时。

第四步, 计算星地时差。利用时间校准系统, 比较星载数据管理计算机与地面 GPS 时间的时差, 即为星时恢复过程产生的星时误差。

测试结果表明:

1) 星载数据管理计算机可以通过载荷计算机自主恢复星时, 星时恢复方案可行;

2) 恢复后的星时与地面标准时间的差值为 $240 \mu s$ 。

此星时恢复方案, 恢复过程产生的误差主要有两部分, 即 Δ_1 和 Δ_2 , 其中 Δ_1 是星载数据管理计算机产生的, Δ_2 是载荷计算机产生的, 这两部分误差均与计算机软件操作过程相关, 可以通过优化软件设计以及仿真分析逐步减小误差。同时, 外部时钟源的精度、星载数据管理计算机复位过程持续时间均会对星时恢复误差产生影响。

此星时恢复方案优点在于: 星载数据管理计算机出现复位或切换备份操作时, 能否自主恢复星时, 保证了卫星系统星时连续性, 提高了卫星可用性, 并且能够将星时恢复误差控制在微秒级, 精度能够满足使用要求。

4 几点说明

1) 星时用户选取。不同的星时用户对星时精度要求不同, 选取对星时精度要求高的星时用户, 有助于提高星时自动恢复的时间精度。考虑系统备份, 在不影响系统工作前提下可以选取多个星时用户用于星时恢复。

2) 采用此方法实现星时自主恢复的前提是星载数据管理计算机与所选取的星时用户使用的时钟同源或者在星载

数据管理计算机复位或切机开始至星时自动恢复这段时间内星时用户时钟产生的误差可以忽略。

3) 在制定星时自动恢复协议时, 需要确定 ΔT_1 、 ΔT_2 和 T_d , 并且要求 $T_d > \Delta T_1 + \Delta T_2$ 。一般要求 T_d 不大于 500 ms, 这样有利于尽快恢复星时。

4) 星时恢复过程中的延时修正量 Δt_1 和 Δt_2 需要分别在星载数据管理计算机调试及星时用户设备调试过程中标定, 标定的准确与否决定了星时自动恢复的误差大小。

5) 星时恢复效果可以通过星载数据管理计算机与星时用户设备间联合测试进行验证。测试时需要具备星地时延、星地时差测量系统, 能够通过集中校时实现对星时的准确校正。

5 结束语

本文提出的基于 1553B 总线的卫星星时高精度自主恢复设计, 能够有效减少对地面人工干预的依赖, 并能够有效的保证在意外情况下星载数据管理计算机工作的连续性。此星时自主恢复方法能够实现微秒级误差, 精度满足工程需要。星时高精度自主恢复设计没有带来额外的硬件开销, 需要增加的仅仅是相关总线协议的制定以及软件开发工作, 在实际工程中具有一定的实用价值。在星载数据管理计算机需要复位或切机的情况下, 大大提高了卫星系统的可用性。

(上接第 153 页)

系数进行适当的降低以节省集群存储资源和减少因副本一致性维护所带来的系统开销。实验结果表明, 在数据访问效率和集群存储资源利用率两方面, 本文提出的基于热点数据块的动态副本调整策略与 HDFS 固定副本策略和基于文件的动态副本策略相比有了明显提升。下一步, 将对增加的副本的数据一致性维护进行深入研究, 以在尽量减少系统消耗的基础上保证数据的一致性。

参考文献:

- [1] 肖中正, 陈宁江, 等. 一种基于文件支持度的动态副本管理机制 [J]. 计算机研究与发展, 2016, 53 (2): 431-442.
- [2] 徐奕奕, 唐培和. 基于分数阶 Fourier 变化的云存储系统重复数据删除算法 [J]. 计算机科学, 2015, 42 (7): 174-177.
- [3] 陈波, 沈炜. 基于 HDFS 的动态副本策略设计与实现 [J]. 工业控制计算机, 2015, 28 (2): 103-105.
- [4] 吴修国. 云计算环境下面向最小成本的数据副本策略 [J]. 计算机科学, 2014, 41 (10): 154-159.
- [5] 邓见光, 赵跃龙. 云计算环境下的动态数据副本管理策略 [J]. 华中科技大学学报 (自然科学版), 2015, 43 (4): 138-143.
- [6] Grace R K, Manimegalai R. Dynamic replica placement and selection strategies in data grids—a comprehensive survey [J]. Journal of Parallel and Distributed Computing, 2014, 74 (2): 2099-2108.
- [7] 张倩, 郑焱, 王嵩. 基于热度的 Hadoop 快速副本复制

参考文献:

- [1] 陈宜元. 卫星无线电测控技术 (下) [M]. 北京: 宇航出版社, 2007.
- [2] 童伟. 基于北斗 II 代/GPS 的电力系统双模时间同步时钟的研制 [J]. 电力建设, 2014, 35 (4): 81-85.
- [3] 王伟, 陈丹, 徐晓光. 航天器自主高精度时间管理系统设计 [J]. 传感器与微系统, 2017, 36 (7): 127-129.
- [4] 杨天社, 李怀祖. 在轨卫星与地面时钟精确同步方法研究 [J]. 系统工程与电子技术, 2002, 24 (5): 103-105.
- [5] 高建军, 兰天, 王斌. 一种高可靠的星载软件时间管理方法 [J]. 计算机测量与控制, 2013, 21 (3): 806-808.
- [6] 穆强. 星载嵌入式软件中断响应时间的分析 [J]. 计算机测量与控制, 2012, 20 (2): 536-537.
- [7] 谭维织, 顾莹琦. 空间数据系统 [M]. 北京: 中国科学技术出版社, 2004.
- [8] 叶志玲, 张猛, 郭坚, 等. 嫦娥一号卫星数据高可靠性保护设计 [J]. 航天器工程, 2008, 17 (1): 53-56.
- [9] 田贺祥, 王同桓, 李璇, 等. 遥感卫星星上时间管理方法 [J]. 传感器与微系统, 2013, 32 (4): 80-82.
- [10] 王文平, 元勇, 王颖, 等. 实践十号返回卫星时间同步系统设计与验证 [J]. 计算机测量与控制, 2017, 25 (8): 237-240.
- [11] Tan P X, Chen Y, et al. Secure fault-tolerant code for cloud storage [J]. Journal on Communications, 2014, 35 (3): 109-114.
- [12] 韩敬峰. 云计算环境下海量数据低负荷存储方法研究 [J]. 计算机仿真, 2016, 33 (4): 390-394.
- [13] 蒋浩. 分布式文件存储系统副本管理方法研究 [D]. 成都: 电子科技大学, 2013.
- [14] 周小玉. HDFS 分布式文件系统存储策略研究 [D]. 成都: 电子科技大学, 2015.
- [15] 李功丽, 赵晓焱, 刘慧. 一种云计算数据副本动态管理策略 [J]. 河南师范大学学报 (自然科学版), 2014, 34 (S2): 130-134.
- [16] 王政英, 于炯, 等. 基于用户访问特征的云存储副本动态管理节能策略 [J]. 计算机应用, 2014, 34 (8): 2256-2259, 2290.
- [17] 饶磊, 杨凡德, 等. 基于热度分析的动态副本创建算法 [J]. 计算机应用, 2014, 34 (S2): 130-134.
- [18] Dong B, Zheng Q, Tian F, et al. An optimized approach for storing and accessing small files on cloud storage [J]. Journal of Network and Computer Applications, 2012, 35 (6): 1847-1862.
- [19] Yan C, Li T, Huang Y, et al. Hmfs: Efficient Support of Small Files Processing over HDFS, Algorithms and Architectures for Parallel Processing [M]. Springer International Publishing, 2014, 54-67.