

7-2-2020

All-digital Simulation Technique with Application to On-orbit Spacecraft Maintenance and Servicing

Gaoxiang Ouyang

1. Academy of Opto-Electronics, Chinese Academy of Sciences, Beijing 100094, China;;

Li Xin

1. Academy of Opto-Electronics, Chinese Academy of Sciences, Beijing 100094, China;;2. University of Chinese Academy of Sciences, Beijing 100049, China;

Dong Xin

1. Academy of Opto-Electronics, Chinese Academy of Sciences, Beijing 100094, China;;

Chengmin Sun

1. Academy of Opto-Electronics, Chinese Academy of Sciences, Beijing 100094, China;;

See next page for additional authors

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research](#), [Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

All-digital Simulation Technique with Application to On-orbit Spacecraft Maintenance and Servicing

Abstract

Abstract: Aiming at the demand for the on-orbit maintenance, a new simulation technique was developed. The simulation contained on-orbit failures reproduction and location, onboard software upgrade and patch. A *virtual technology* was used to construct all virtual hardware environment, imitate their function and simulate the real satellite control system. This all-digital platform implemented precise multi-type faults injection, internal state monitoring and data stream intervention very well. So it could roundly simulate and test the software upgrade of the attitude and orbit control systems. It could also be used in the high-fidelity simulation of satellite-ground synchronous operation and maintenance, especially it is very helpful to improve the reliability of a satellite and prolong its life.

Keywords

virtual technology, all-digital platform, attitude and orbit control, on-orbit maintenance, synchronous operation

Authors

Gaoxiang Ouyang, Li Xin, Dong Xin, Chengmin Sun, and Yang Xin

Recommended Citation

Ouyang Gaoxiang, Li Xin, Dong Xin, Sun Chengmin, Yang Xin. All-digital Simulation Technique with Application to On-orbit Spacecraft Maintenance and Servicing[J]. Journal of System Simulation, 2016, 28(4): 979-990.

全数字仿真及其在卫星在轨维护中的应用

欧阳高翔¹, 李鑫^{1,2}, 董鑫¹, 孙成明¹, 杨新¹

(1. 中国科学院光电研究院, 北京 100094; 2. 中国科学院大学, 北京 100049)

摘要: 针对长寿命高价值卫星在轨维护的实际需求, 开展卫星姿轨控分系统在轨故障复现、定位与星载软件升级、补丁快速仿真验证的新方法研究。采用虚拟化技术构造星载控制系统运行所需的全部硬件环境及其外部数据源, 可等效模拟整星控制系统的实际硬件功能, 进而仿真星上控制系统的空间运行状态。这种全数字化平台具备了多类型故障精确注入、内部状态监控和全数据流干预等特点, 可对卫星姿轨控系统在轨维护中的软件升级行为仿真和全覆盖测试验证。该全数字化平台还能应用于卫星在轨运维的星地高保真同步运行, 尤其是对卫星运行末期的可靠性增长和有效生命周期延长起到了重要的支撑保障作用。

关键词: 虚拟化; 全数字化平台; 姿轨控; 在轨维护; 同步运行

中图分类号: V19

文献标识码: A

文章编号: 1004-731X(2016)04-0979-12

All-digital Simulation Technique with Application to On-orbit Spacecraft Maintenance and Servicing

Ouyang Gaoxiang¹, Li Xin^{1,2}, Dong Xin¹, Sun Chengmin¹, Yang Xin¹(1. Academy of Opto-Electronics, Chinese Academy of Sciences, Beijing 100094, China;
2. University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Aiming at the demand for the on-orbit maintenance, a new simulation technique was developed. The simulation contained on-orbit failures reproduction and location, onboard software upgrade and patch. A *virtual technology* was used to construct all virtual hardware environment, imitate their function and simulate the real satellite control system. This all-digital platform implemented precise multi-type faults injection, internal state monitoring and data stream intervention very well. So it could roundly simulate and test the software upgrade of the attitude and orbit control systems. It could also be used in the high-fidelity simulation of satellite-ground synchronous operation and maintenance, especially it is very helpful to improve the reliability of a satellite and prolong its life.

Keywords: virtual technology; all-digital platform; attitude and orbit control; on-orbit maintenance; synchronous operation

引言

高价值卫星通常要求较长的生命周期, 诸如地球同步卫星一般设计寿命为 8~10 年。然而, 因卫

星长期暴露在高辐射的外太空环境中频繁地过冷过热, 极易引起整星部件的工作能力逐渐丧失直至失效, 在卫星生命末期阶段尤为突出。为了延长卫星的有效服务时间, 目前包括在轨燃料加注、失效部件替换和星载软件补丁与升级等在轨维护得到了极大的重视^[1], 与之相关的技术也已成为研究热点。考虑到现有的技术条件和成本, 当前国内在轨维护还多集中在星载软件的升级与补丁, 以及在



收稿日期: 2014-12-01 修回日期: 2015-01-15;
基金项目: 国家自然科学基金(61308101);
作者简介: 欧阳高翔(1977-), 男, 四川, 博士, 高工, 研究方向为卫星总体和姿轨控设计与仿真; 李鑫(1990-), 男, 山西, 硕士生, 研究方向为复杂航天器姿态控制。

<http://www.china-simulation.com>

轨故障复现与定位后的冗余部件指令切换^[2]。通常卫星的部件问题都会通过星上姿轨控分系统表征出来,如姿态超差或上注指令不响应等。另外,星上姿轨控系统本质是一类嵌入式系统,其内置软件是整星控制算法和飞控任务的具体承载者^[3]。因此,在轨服务的主体对象多以星上控制分系统为主,具体技术手段则是通过对其内置软件升级来尽可能地弥补或减小整星组件失效对飞控任务的影响或纠正软件错误。

卫星在轨维护实施之前,均需要通过地面仿真系统进行验证确认。这类验证多是通过在地面构建与之相配套的半物理仿真系统^[4],开展升级或补丁软件对原系统功能集的影响范围确认。然而,由于库存数量和厂商的升级换代,某些星上硬件可能断货或停产造成实际硬件的缺少,使得在地面上基于半物理仿真系统的软件升级将很难创建软件补丁。另外,对于某些关键任务下的在轨故障快速隔离与清除,采用半物理仿真系统的测试验证显示出了极大的局限性。为此,需要探寻新的仿真验证方法和平台,以适应当前航天领域针对高价值卫星的低成本、高效和灵活的在轨服务迅捷实施。

与此同时,计算机硬件性能的不提高和软件技术的不断进步使得一种称之为虚拟技术^[5-6]的全数字仿真方法开始应用于嵌入式系统研制的各阶段中^[7],有效地缓解了上述问题,并带来了航天嵌入式系统研制和仿真模式的变革。

1 全数字仿真技术原理

全数字仿真是通过对目标机 CPU、外围芯片、I/O 端口、中断和时钟等虚拟模拟器的组合,在宿主机上构造出目标机内置软件运行所必须的虚拟硬件环境,在功能上与真实硬件环境几乎等效无差异。

1.1 目标机 CPU 处理器模拟

全数字仿真系统的核心部件是目标机 CPU 模拟器,其工作原理是将目标机指令转换为宿主机指令执行,即仿真每条指令在目标处理器上的执行效

果,包括对内存和寄存器等读写操作,完成内置软件在真实硬件上的动态运行过程的模拟。下面以航天领域常用的 SPARC V7 架构 TSC695F 为对象进行关键技术说明。

TSC695F 处理器数字虚拟化主要基于精确指令周期边界的指令集体系统结构模型(Instruction Set Architecture),模拟内置软件运行过程中对处理器的寄存器、标志、内存等的变化影响。考虑到 CPU 模拟器仿真速度,并没有模拟各模块间的电信号变化和传递,但同时保证足够的计算精度。

由于 CPU 模拟器是针对指令集的功能性模拟,以获得目标机程序正确运行结果为目标,而忽略目标机指令的执行细节。为此,处理器计算模型仅是真实地仿真出指令集中每条指令的取指、译码和执行的过程,以及由此引起的寄存器、CPU 标志、存储器内容数据等变化。计算模型的主体是一个具有 3 步过程的循环体,分别对应取指、译码和执行 3 步操作。整个过程的流程如图 1 所示。

(1) 取指操作:根据指令首地址从虚拟内存器中获取固定 32 位长度的机器指令字节;

(2) 译码操作:根据译码规则过滤指令前缀和识别操作码,并确定机器指令对应汇编格式指令的助记符。基于指令格式解析就能得到机器指令对应汇编格式指令的具体类型^[8],然后调用宿主机对应功能的内置逻辑运算函数,进入第 3 步;

(3) 执行操作:根据操作码后面的机器指令继续分离出操作参数,然后执行机器指令定义的动作并更新相关处理器状态。然后回到第一步,获取下一条指令。

1.2 寄存器模拟

SPARC V7 处理器主要包含 3 种类型处理器的控制/状态寄存器^[9]:1) 整数处理单元 IU 的控制/状态寄存器,2) 浮点处理器 FPU 的控制/状态寄存器,3) 协处理器 CP 的控制/状态寄存器。

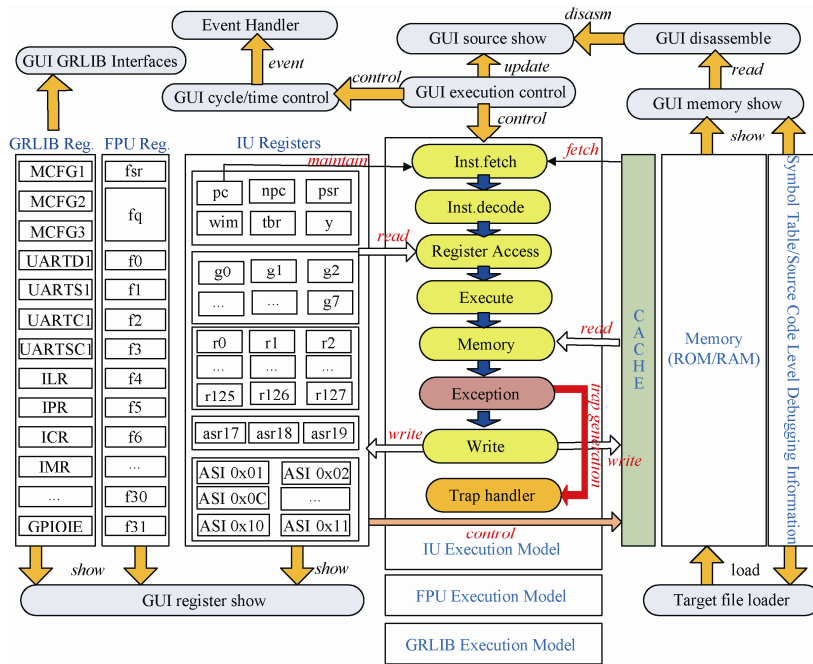


图 1 目标机 CPU 运行流程图

虚拟寄存器被设计为包含一定内存空间大小的结构体容器, 针对实际存储要求采用 byte 类型的 8 维数组, 即可支持 64 位的寄存器。通过自定义方法和逻辑控制, 使得寄存器容器模型可满足不同 CPU 的不同类型寄存器模拟。对于有位意义的寄存器则定义映射器, 用于描述位和位意义的对应关系, 实现不同寄存器对位的不同操作。

1.3 存储器模拟

由于 CPU 和存储器具有较高的耦合度, 且存储器是一个被动数据源。因此, 虚拟存储器模块与指令集模块之间的连接是通过向指令集模块提供回调接口的方式实现。

存储器虚拟模块由两部分组成: 虚拟存储空间和虚拟存储管理机制。虚拟存储空间是实现物理存储空间的模拟, 为实际程序和数据提供存放空间。TSC695F 有 32 根地址线可直接寻址地址空间达到 4G 字节^[10], 考虑到内存占用效率, 虚拟存储空间在实际实现中根据构建系统的寻址范围, 而动态申请宿主机内存空间。

运行在 TSC695F 中的程序使用的都是逻辑地址, 在真正读写存储器前需要将逻辑地址转换成物

理地址。因此, 对 TSC695F 处理器内置存储器在软件模拟时, 基于存储器管理 MMU (Memory Management Unit)原理实现逻辑地址到物理地址的转换计算。

1.4 陷阱中断模拟

TSC695F 支持同步和异步两种陷阱, 也称之为中断。上述两种陷阱模拟单元负责接收和响应中断控制器模拟模块发送的外设中断请求, 并执行中断请求对应的中断处理程序。因此中断机制模拟模块由两部分组成: 陷阱响应模块和陷阱处理模块。

(1) 陷阱响应模块模拟中断过程, 即当有中断请求, 则获取中断类型号, 然后调用中断处理部件执行相应的中断处理程序。

(2) 陷阱处理模块模拟中断向量表和中断处理程序, 用指针映射方式实现表号与处理过程对应。

2 数据交互及接口仿真模拟

除了对目标机数字虚拟化外, 还必须对外设芯片的功能行为进行建模和模拟, 但并不仿真真实芯片的物理电气特性。

2.1 通讯接口模拟

数据通讯接口仿真的核心是对目标机外围芯片模拟,即实现目标机与外部数据激励模型之间数据交换的通讯桥梁或接口单元控制的模拟仿真。星上芯片通讯接口主要分为以下几类^[11]:

- 1) 姿态传感器、执行机构通讯接口模拟,如串口芯片: 8252 等;
- 2) 星务轮询广播通讯接口模拟,如总线芯片: CAN, 1553B 等;
- 3) 星上状态交换接口模拟,如位口芯片;
- 4) 外部中断连接硬件接口模拟,如外部中断控制芯片: 8259 等;
- 5) 遥控、遥测系统接口模拟,如控制芯片: FPGA 板卡。

通讯接口硬件模拟是通过开辟相应的数据缓冲区,模拟数据 FIFO 通讯过程,同时维护其状态寄存器和控制器,按照具体芯片手册编写相应的时序处理逻辑,这里时钟频率统一由虚拟目标机 CPU 提供。按照规定的接口约束提供针对虚拟内核控制单元的回调函数接口,实现与虚拟内核之间的状态触发和数据交互。虚拟目标机可以直接或通过上述虚拟外设与数据激励模型连接,实现数据交换和时钟同步,如图 2 所示。

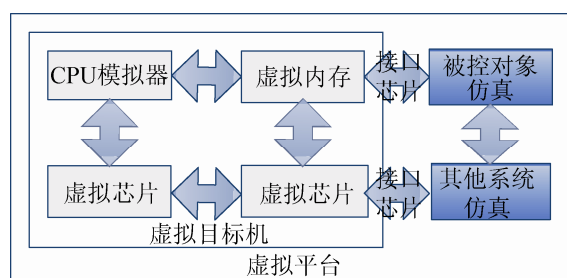


图 2 目标机虚拟化平台

2.2 数据交互通讯模拟

星载控制系统的最大特点是其与外设进行类型多样且高频度的数据交换,由于任何数据及数据交互过程均可被抽象为某种数据交互模型,因此在仿真环境中把该数据交互模型称为数据流。

实际中,星上控制系统收到的信息主要有:卫星工作状态、卫星部件工作状态、卫星部件采集量、遥控指令、注入数据包、星务广播指令等。与此相对应,发送信息主要有:卫星部件开关信息、卫星部件控制信息、遥测数据与重要数据。上述数据流可以理解为按照一定规则映射成的卫星状态标志量、部件标志量、卫星数据信息和部件数据信息的集合。因此,星上数据交互不管以何种方式,不管是对虚拟目标机还是驱动模型,在接口层上接收或发送的内容都是具有某种次序的字节串。数学映射关系如下:

$$f(S) = D, S \in \{float, integer, byte\} \quad (1)$$

式(1)中集合 S 表示了星上控制系统的交互数据类型,通过前面分析可知所有交互数据均可归结到浮点、整型和字符三种类型。集合 D 表示了映射 f 下的有序字节串,而映射 f 是作用在交互数据上的通信协议规则。为此,在对星上控制系统数据双向通讯过程——打包和解包的模拟,只需要将原始数据抽象为 3 类数据类型之一,并基于通讯协议规则将其映射成有序数据流。

3 数据激励模型

对一个真实卫星在轨运行过程的模拟,除了要对星上控制系统硬件虚拟化外,还要将其进行交互的数据激励模型进行数学建模,构成必要的外部数据激励源。

3.1 卫星姿态轨道数学模型

卫星姿态轨道数学模型包含了运动和动力学两部分,通过各自的微分状态方程实现姿轨控数值积分仿真。其姿态动力学与运动学方程为:

$$\begin{aligned} I_s \dot{\omega}_s + \tilde{\omega}_s I_s \omega_s + R_{as1} \dot{\omega}_{a1} + R_{as2} \dot{\omega}_{a2} + F_{s1} \ddot{\eta}_1 + F_{s2} \ddot{\eta}_2 &= T_c + T_d \\ I_{a1} \dot{\omega}_{a1} + R_{as1}^T \dot{\omega}_s + F_{a1} \ddot{\eta}_1 &= T_{a1} \\ I_{a2} \dot{\omega}_{a2} + R_{as2}^T \dot{\omega}_s + F_{s2} \ddot{\eta}_2 &= T_{a2} \\ \ddot{\eta}_1 + 2\xi_1 \Omega_1 \dot{\eta}_1 + \Omega_1^2 \eta_1 + F_{s1}^T \dot{\omega}_s + F_{a1}^T \dot{\omega}_{a1} &= 0 \\ \ddot{\eta}_2 + 2\xi_2 \Omega_2 \dot{\eta}_2 + \Omega_2^2 \eta_2 + F_{s2}^T \dot{\omega}_s + F_{a2}^T \dot{\omega}_{a2} &= 0 \end{aligned} \quad (2)$$

式(2)中的变量定义为: I_s 为航天器惯量阵; I_{a1}, I_{a2}

为帆板惯量阵; ω_s 为航天器绝对角速度矢量; ω_{a1} , ω_{a2} 为左右太阳帆板相对于航天器的转速; Ω_1, Ω_2 为左右太阳帆板模态频率阵; η_1, η_2 为左右太阳帆板的模态坐标矢量; F_{s1}, F_{s2} 为帆板振动与星体转动耦合系数阵; F_{as1}, F_{as2} 为帆板振动与帆板转动耦合系数阵; R_{a1}, R_{a2} 为帆板驱动与星体转动耦合系数阵; T_c 为作用于航天器上的控制力矩矢量; T_d 为作用于航天器上的扰动力矩矢量; T_{a1}, T_{a2} 为作用于左右太阳帆板的驱动力矩矢量。姿态动力学数学计算公式为:

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = 1/2 \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \begin{bmatrix} \omega_x - \omega_{ox} \\ \omega_y - \omega_{oy} \\ \omega_z - \omega_{oz} \end{bmatrix} \quad (3)$$

式(3)中: $\omega_s = (\omega_{ox}, \omega_{oy}, \omega_{oz})^T$ 为星体在本体系下的绝对角速度, 由式(2)右端函数计算得到。另外, 参考坐标系选为轨道坐标系时, 由于该参考坐标系是一个动坐标系, 因此需要考虑轨道角速度在本体系的投影分量 $[\omega_{ox}, \omega_{oy}, \omega_{oz}]$, 一般而言卫星所在的轨道大多为圆轨道, 其计算公式为:

$$\begin{bmatrix} \omega_{ox} \\ \omega_{oy} \\ \omega_{oz} \end{bmatrix} = C_{bo}(\bar{q}) \begin{bmatrix} 0 \\ -\omega_o \\ 0 \end{bmatrix} \quad (4)$$

通过对微分状态方程(3)进行积分计算, 可以得到积分时刻的四元数 q , 于是姿态阵 C_{bo} 可由上述四元数 q 获得:

$$C_{bo} = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_1q_2 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_1q_4 + q_2q_3) \\ 2(q_1q_3 + q_2q_4) & 2(-q_1q_4 + q_2q_3) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \quad (5)$$

将式(5)按照 3-1-2 转序可反解出卫星三轴欧拉姿态角^[12]。

为了能同时处理圆轨道和零倾角轨道, 其外推计算模型采用笛卡尔坐标方式, 计算公式为:

$$\begin{cases} \dot{\bar{R}} = \bar{V} \\ \dot{\bar{V}} = -\mu_e \frac{\bar{R}}{R^3} + \bar{P}(R, V, t) + F \end{cases} \quad (6)$$

式(6)中 \bar{P} 为卫星受到的摄动加速度, 即:

$$\bar{P} = \bar{P}_G + \bar{P}_D + \bar{P}_R + \bar{P}_S + \bar{P}_M \quad (7)$$

其中, \bar{P}_G 为地球非球形引力摄动加速度; \bar{P}_D 为大气阻力摄动加速度; \bar{P}_R 为太阳光压摄动加速度; \bar{P}_S 为太阳对卫星的引力摄动加速度; \bar{P}_M 为月球对卫星的引力摄动加速度, 上述摄动项的具体计算公式参见文献[13]。 F 为推进系统的推力, 包括轨控发动机与姿控发动机的推力。

姿态&轨道联合仿真计算模型如图 3 所示。

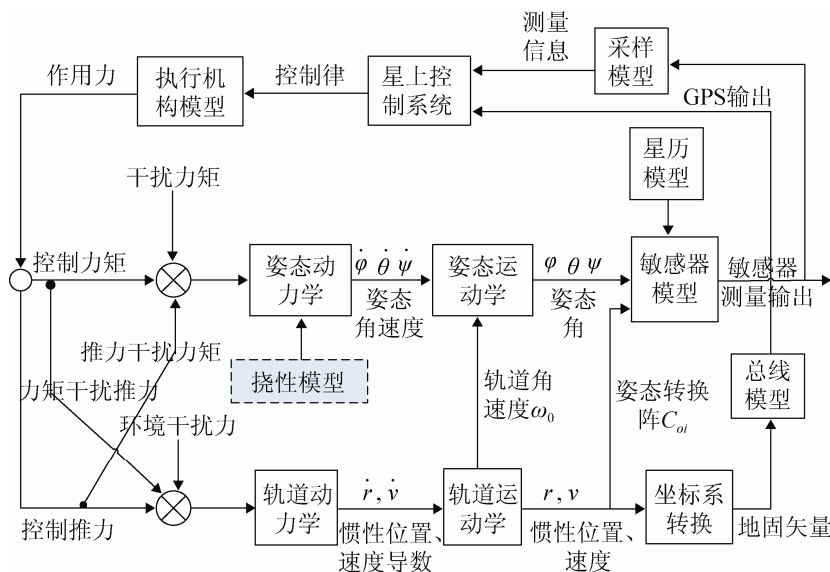


图 3 姿态轨道耦合仿真流程图

需要强调的是式(3)和式(4)中通过轨道角速度 ω_o ，轨道参考坐标系转换矩阵 C_{bo} ，以及推力干扰力矩、力矩干扰推力项将姿态和轨道仿真耦合在一起，使得整星运行仿真更符合实际情况。

3.2 卫星典型部件模型

除此而外，卫星数学模型中还包含了典型的部件模型如传感器和执行机构。传感器一般来说都是以某个天体作为参考测量基准，如地敏是以地心作为测量基准，在数学建模时只需要将地心矢量 E_o 投影到地敏安装坐标系即为地敏输出。同理，星敏的测量输出是星敏光轴和横轴在惯性空间下的矢量指向，太敏是太阳矢量方向在其安装坐标系中的投影，磁强计是地磁矢量在磁强计安装坐标系中的投影。惯性器件则是基于式(2)计算得到的绝对角速度 ω_s 在本体系中的投影。上述投影计算都需要采用式(5)的坐标系转换矩阵 C_{bo} 。

执行机构作用方式包含力矩和推力两种类型。力矩输出执行机构模型是基于角动量守恒原理，通过改变转子机构转速(动量轮)和方向(框架力矩陀螺)实现对卫星本体的绝对角速度变化，亦即式(2)所示。推力输出模型则是仅考虑惰性环节的常值输出。执行机构输出将分别是式(2)和式(6)右边的 T_c 和 F 两项。

3.3 GPS 信号广播模拟

GPS 数据广播是卫星在轨运维的常规措施，用以纠正星上轨道外推累计误差和外部摄动干扰引起的实际轨道变化。对 GPS 数据广播仿真便是模拟实际卫星在轨运行过程中的轨道参数定时修正行为，进而保证卫星高精度对地定向。考虑到 GPS 数据是定义在 WGS-84 坐标系下，采用简化仿真模型将 GPS 信号处理为包含精确时戳的卫星地固坐标系位置和速度信息。为此，需要将地面卫星轨道计算模型中 J2000 惯性坐标系下的位置 R 和速度 V 投影到相应时刻的地固坐标系中。上述坐标转换计算公式为：

$$\bar{R} = (HG)\bar{r} \quad (8)$$

式(8)中 $(HG)=(EP)(ER)(GR)$ ，(EP)、(ER)、(GR)分别是极移、自转和岁差章动矩阵，具体计算公式参见文献[13]。

3.4 卫星运行空间环境模拟

卫星运行的外太空环境对卫星的精确仿真也是至关重要的，还需要对包括星历计算模型、地磁模型等外围环境进行高精度数学建模。

太阳星历计算模型采用平根数解析公式：

$$us = SUNM_0 + 0.98561 \times (UTC_k - UTC_0) / 86400 + 1.9139 \times \sin(2\pi \times (UTC_k - UTC_0) / 86400 / 365.25) \quad (9)$$

$$S_{ix} = \cos(us)$$

$$S_{iy} = \cos(23.4452 \times \pi / 180) \times \sin(us)$$

$$S_{iz} = \sin(23.4452 \times \pi / 180) \times \sin(us)$$

式(9)中变量 UTC_0 为计时零时刻时间； $SUNM_0$ 为相对于计时零时刻 UTC_0 的太阳平近点角； UTC_k 为当前时刻 t_k 的 UTC 时间； S_i 为惯性坐标系下的太阳单位矢量； us 为当前时刻黄赤交线与 J2000 坐标系 X_i 的夹角。

地磁计算模型采用 GRIM4-S4 地球磁场模型，对于一般卫星，磁场的阶数和维数仅需要取到(8×8)。

3.5 时间系统模拟

数字化仿真平台的整个系统运行在统一的虚拟时间域中，所有虚拟设备基于 CPU 单一时钟源而保持时间同步。在仿真硬件和目标软件中可以引入基于虚拟时间的延时，虚拟时间通过输出给外部虚拟部件用于同步和集成。另外，虚拟时间域可以放慢来适应外部较慢的工具或实物设备。

在整个仿真计算中时间是独立单元，但是在计算不同物理量时，却使用不同时间系统^[13]。在计算恒星时用世界时 UT1；数学仿真积分系统采用地球动力学时 TDT；岁差和章动量的计算采用 TDB 时等。所以数字仿真系统内部采用一个时间系统来维护仿真时间基准，以及各时间系统之间的转换。下面给出各种时间系统的定义以及它们之间的转换公式。相互关系如下：

$$\begin{aligned} UT1 &= UTC + \Delta UT1, TAI = UTC + \Delta AT \\ TDT &= TAI + 32^s.184, TDB = TDT + \Delta TD \end{aligned} \quad (10)$$

式(10)中 $\Delta UT1$ 可从地球自转参数文件^[14]中获得。时间系统基准则采用星箭分离时刻儒略日, 另以卫星的 GPS 记时时刻为仿真计算基准, 相对于基准时刻的偏移量则作为虚拟 CPU 的计时器数值, 并由其统一维护时间的增减操作。

4 全数字仿真平台应用

4.1 数字化仿真系统构建

考虑到不同卫星平台在其系统组成上存在较大差异, 将采用配置方式定制化特定目标虚拟系统。另外, 将仿真平台配置现场保存可在以后加载, 实现对关键场景对不同输入条件下全方位仿真验证

证。构建过程概括为 3 个步骤:

首先, 从仿真模型库中选取所需的 CPU 模拟器、虚拟芯片等标准模型, 若当前模型库中没有所需的虚拟芯片, 则按照芯片手册和编程接口约束定制特殊虚拟芯片。

其次, 进行系统配置参数设置, 包括系统使用的 CPU 类型、CPU 主频、内存大小、使用的虚拟芯片、虚拟芯片的端口地址、连接关系等。

最后, 根据整星控制分系统设计方案, 从标准化模型库中配置卫星系统部件, 包括执行机构和敏感器的选型, 安装矩阵和扰动源类型等基本参数设置。

通过对采用文件配置和自定义开发获得整个基于数字化平台的整星仿真系统, 如图 4 所示。

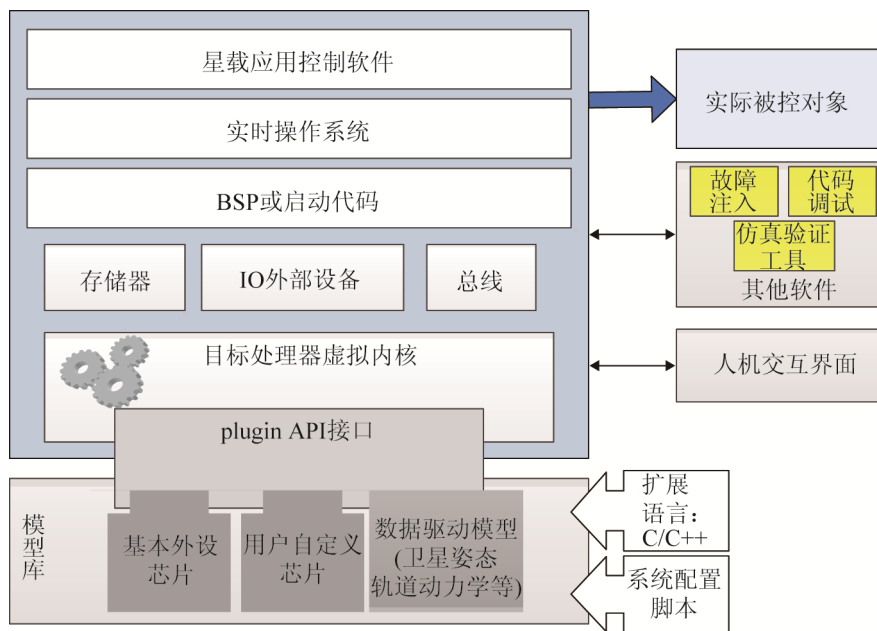


图 4 数字化平台组成框架

上述虚拟系统构成完整闭环回路, 被测嵌入式软件在虚拟化目标机上执行。受控对象则采用数学模型作为数据激励源。以上构成的全数字化测试平台在功能上同真实物理平台是等效的。

在实际虚拟系统使用过程中, 星载软件在不做任何修改下编译形成目标码二进制文件。仿真框架核心控制单元根据配置文件初始化数字化平台工

作环境, 并通过一个加载器将该目标码文件放置到 CPU 模拟器的虚拟 ROM 内存中。具体内容包括物理存储空间的申请和初始化以及代码区、数据区和寄存器区的初始化。在模拟器启动后就进入相应的 CPU 指令集运行模拟模块, 并同时接受用户交互操作, 启动流程如图 5 所示。

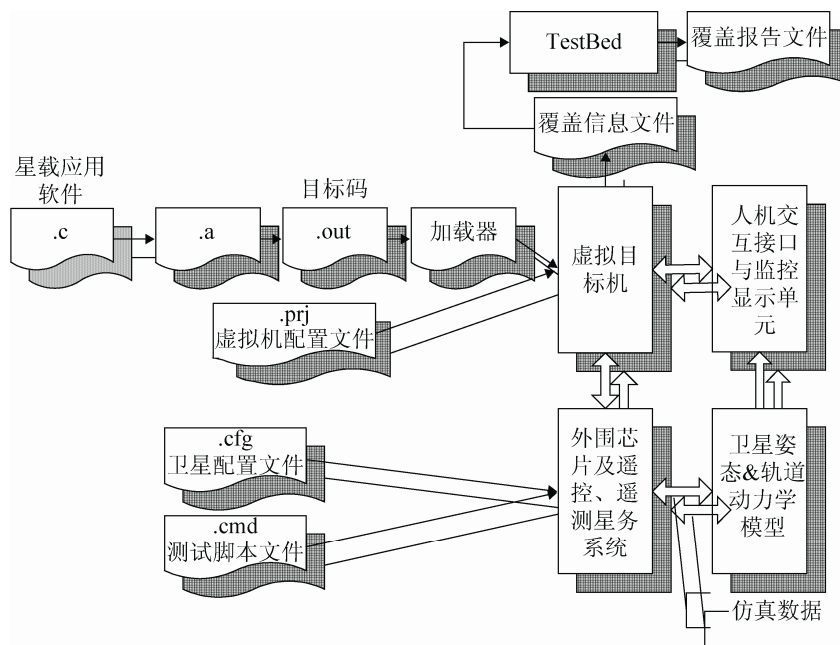


图 5 数字化仿真平台启动过程

通过提供目标机模拟和对象仿真模型库以及二者之间的数据和控制接口，在单台 PC 机上就可进行闭环仿真，大大降低了集成代价。同时由于目标机和外围设备的虚拟化数字平台可复制使用，使得测试不受实际测试设备可用性的限制。

在仿真验证过程中可立即对被测软件的输出同受控对象仿真实输出结果进行比较，从而有效验证升级或补丁软件的正确性。通过调整受控对象仿真状态，人为加入扰动等“干预”，考验上述被测软件的可靠性和边界约束，对其功能进行全方位仿真验证。

4.2 状态监控

在实际卫星故障定位中，将数字化平台的寄存器和存储器等内部状态值设置为故障发生前的状态，同时断点在故障发生时的程序执行处，查看虚拟目标机寄存器和内存单元的敏感量变化值，可实现星上故障的快速复现和定位。上述过程则需要借助在线调试技术手段支持。

在数字化平台实现中指令是逐条被模拟执行的，通过在每条指令执行之前检查中断表中在当前时间是否有需要执行的断点设置，如果有断点则触发宿主机中断进行调试，否则就继续正常顺序执

行。单步调试算法流程如下：

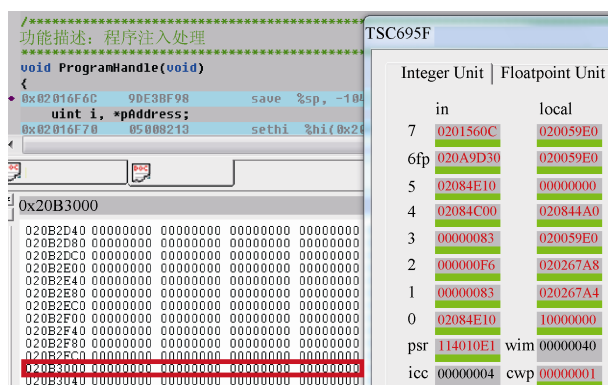
- 1) 从被调试程序的当前栈帧的栈帧信息结构体指针获取指令 pc 值，并基于符号表信息找到对应源程序程序行及其指令地址空间范围；
- 2) 与断点表中断点信息比较，如果存在相同设置则激发被调试应用程序的宿主机单步陷入功能，同时将栈帧信息中关于内存和寄存器中信息加以显示；
- 3) 等待被调试应用程序执行完一条宿主机陷入事件后，读出被调试程序当前陷入位置的指令寄存器值，记为 TRAP PC；
- 4) 判断 TRAP PC 当前指令是否为 call 指令，若是则跳转到第 6 步执行；否则顺序执行；
- 5) 如果 TRAP PC 仍在当前行对应的段地址空间范围内，则回到第 3 步执行；否则根据 TRAP PC 的值采用与第 1 步类似的方法，找到地址范围包含 TRAP PC 所在的行并记为 line，取消被调试程序的指令级单步执行的属性，根据行号 line 指示执行到的位置；跳转到第 7 步；
- 6) 取消被调试程序的指令级单步执行属性；在 call 指令返回地址插入保存断点信息，让被调试应用程序从断点位置处继续执行；
- 7) 结束单步调试。

借助上述单步调试可比较单条指令执行前后寄存器和存储器内部状态, 实现指令逐条执行情况的监控。同时, 也可通过修改寄存器和内存单元的内容, 间接或者直接地改变整个数据流的运行路径, 实现对控制系统运行状态的全程监控。

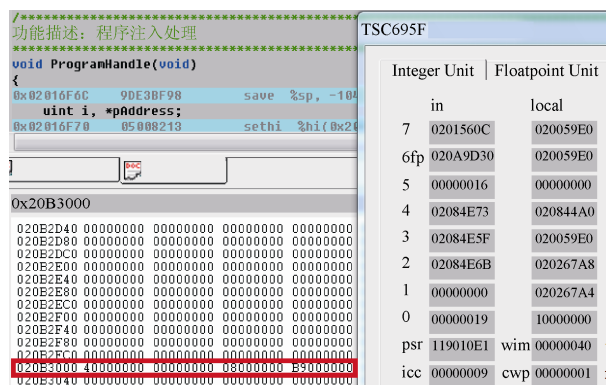
4.3 在轨维护实际应用

实际中, 星载姿轨控系统在轨维护的重要形式是对其内置软件的替换或升级。通过一种被称为 Hook 技术^[14]来定义函数指针并实现其接口函数, 可在不修改原软件系统框架和代码的情况下, 实现用户定制的升级或替代功能消除硬件缺陷造成的错误影响。下面给出一个具体的星载软件升级应用场景。

某高轨卫星在寿命末期阶段若正常工作陀螺个数减至 4 个以下时, 星上原始陀螺故障算法将不能基于平衡方程进行故障陀螺识别与剔除。为此, 将采用其他姿态敏感器经过差分处理后进行辅助识别, 需要替换原有的星上陀螺故障诊断程序。假定星载控制计算机程序运行存储单元 RAM 起始地址为 0x02000000, 其中程序注入区为 0x20B3000~0x20C2FFF 共计 64 k。将替换的陀螺故障诊断新程序首先注入到程序注入区, 并在 Hook 模块中设定被修改的模块 ID 和注入模块的起始地址, 星载应用软件就将不再执行原来 RAM 区的函数模块, 而执行上传注入的模块。图 6 给出了上述补丁程序上注行为过程仿真。



(a) 注入前程序区内存值



(b) 注入后程序区内存值

图 6 程序注入内存显示结果图

上注代码片段及对应指令二进制编码如下:

```
call 0x20452f4 40 00 08 B9
```

补丁代码注入的地面仿真中借助在线调试的内存查看机制, 可获知起始地址为 0x20B3000 的内存单元初始值为 0, 注入补丁后对应内存地址单元值变更为 0x40,0x00,0x08,0xB9, 意味着程序将会跳转到入口地址为 0x20452f4 的函数模块中, 跳转过程的指令译码执行如图 7 所示。

上述在轨维护实际操作过程中在 RAM 代码注入区内内存地址为 0x20B3000 中插入新的程序段, 会从原来代码执行位置跳转到入口地址为 0x20452F4 的 NEW_GyroDiagnose 函数模块, 而替

换 RAM 区地址 0x2005F98 后面 GyroDiagnose 代码模块, 实现对星上软件行为更新, 完成在轨维护的软件补丁行为。在补丁程序上注前都需要通过地面仿真验证, 评估补丁程序对其他功能模块的影响。由于数字化验证平台全部硬件以软件方式模拟, 可通过参看虚拟内存、寄存器和在线调试等手段全程监控补丁软件的修正行为。借助数字化平台监测和控制手段, 上述仿真测试可以是系统级的, 即不仅具有测试上注软件的新功能, 而且还可测试上述新增或修改对整个卫星控制过程的影响。因此, 补丁软件在真正上注卫星之前, 地面能够使用全数字仿真平台模拟其上注行为进行全方位评估。

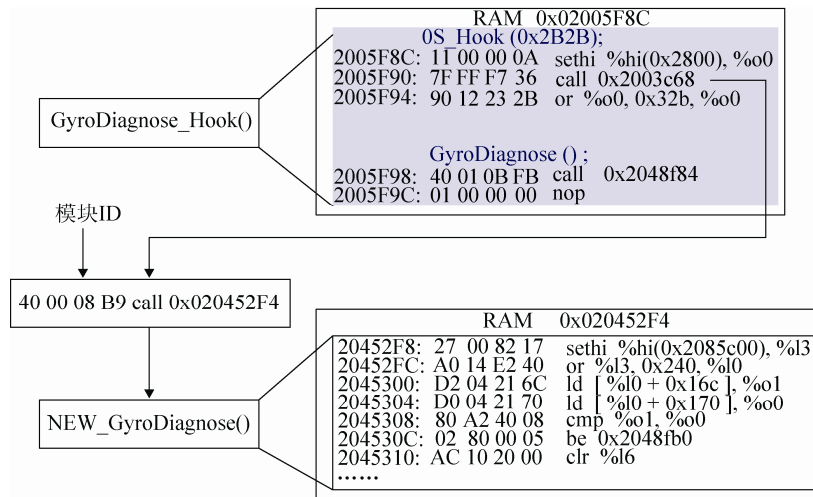


图 7 新的陀螺故障诊断程序在轨注入

4.4 星地同步高保真仿真

由于数字仿真平台运行在统一的虚拟时间域中，因此可以调整仿真时间尺度实现星地同步仿真，这对于在轨运维提供了重要的支撑平台。下面给出运行在低圆轨道上某小卫星入轨转正常对地模式的星地同步仿真过程，其中卫星系统的基本配置参数如下：

$$I_s = \text{diag} \{77.0 \quad 146.6 \quad 182.35\},$$

$$\text{WheelM} = \begin{bmatrix} 1 & 0 & 0 & -0.5774 \\ 0 & 1 & 0 & -0.5774 \\ 0 & 0 & 1 & -0.5774 \end{bmatrix}, \quad (11)$$

$$\text{GMMat} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

式(11)中 I_s 为整星三轴主惯量，单位 kg/m^2 ， WheelM ， GMMat 分别为动量轮、陀螺和磁力矩器安装阵。卫星入轨前俯仰方向上的动量轮始终维持常数旋转，由于陀螺效应而保持定向性，入轨后采用 B-dot 消旋控制律^[16]进行速度阻尼和姿态控制。转正常对地后采用 PD 零动量控制方式进行航天器对地三轴控制，其中俯仰通道 PD 控制律计算式为：

$$k_{p2} = \frac{T_d}{\theta_{ss}} - 3n^2(I_x - I_z)$$

$$k_{d2} = 2\xi \cdot I_y \sqrt{\frac{T_d}{\theta_{ss} \cdot I_y}} \quad (12)$$

式(12)中的变量 T_d ， θ_{ss} ， R_e ， h ， μ 和 ξ 分别为干扰

力矩、期望姿态指向精度、地球半径、轨道高度、地球引力常数和阻尼比，这里轨道高度为 543.66 km，其最大常值干扰力矩为 $500 \mu\text{Nm}$ ，期望指向精度为 10^{-3} 度，阻尼比 ξ 取值 0.707。

滚动和偏航通道 PD 控制参数计算公式为：

$$k_{p1} = \frac{1 - (\theta_{ss} / T_d) \cdot a}{\theta_{ss} / T_d}, k_{p2} = \frac{1 - (\theta_{ss} / T_d) \cdot c}{\theta_{ss} / T_d}$$

$$k_{d1} = \sqrt{\frac{8uI_x^2 I_z^2 \xi^2 + 4v\xi^2 I_x I_z - 16\xi^4 uI_x^2 I_z^2}{I_z^2 + I_x w^2 + 2I_z I_x w - 4w\xi^2 I_x I_z}}, \quad (13)$$

$$k_{d3} = w \cdot k_{d1}$$

式(13)中变量 u ， v ， w 计算公式分别如下：

$$u = \sqrt{\frac{1}{I_x I_y} (k_{p1} c + k_{p1} k_{p3} + ac + ak_{p3})}$$

$$v = I_x c + I_x k_{p3} + aI_z + k_{p1} I_z + b^2 \quad (14)$$

$$w = \frac{-uI_z + c + k_{p3}}{uI_x - k_{p1} - a}$$

式(12)、(13)和(14)中变量 a ， b ， c 和 n 计算公式分别是：

$$a = 4n^2 \cdot (I_y - I_z), b = -n \cdot (I_x - I_y + I_z)$$

$$c = n^2 \cdot (I_y - I_x), n = \sqrt{\frac{\mu}{(R_e + h)^3}} \quad (15)$$

据此计算得到 PD 控制律参数值 $k_p = [1.3813, 3.0157, 3.0495]$ ， $k_d = [78.7652, 255.7030, 273.5815]$ ，图 8 给出了数学和数字两类仿真入轨后的航天器对地三轴姿态变化控制过程。

从图 8 中的仿真结果可明显看出上述 2 类仿真中俯仰角变化频率均逐渐减小, 并同时滚动和偏航轴上出现进动, 即速度阻尼过程。一旦捕获太阳后星上控制系统启动其他 3 个动量轮进入零动量控制方式, 三轴姿态迅速控制到零附近, 即正常对地模式。上述 2 种仿真的动态过程基本一致, 数学与数字仿真的差异在于: 数字仿真中引入的虚拟目标机造成了计算模型精度误差和延时性。数字仿真中借助定时器模拟串口通讯过程, 造成采样数据并非当前周期内实时数值而带来时延, 因此数字仿真也更真实地反映实际控制过程。当受控对象传递函数具有谐振模态或高频模态时, 延时情况会造成数字和数学仿真上的更大差异, 图 8 也部分反映了上

述现象。

另外, 若数字仿真平台与 STK^[17]等可视化软件进行数据连接, 可同步显示卫星在空间的几何位置关系和自身姿态变化。上述小卫星入轨转对地过程的三维同步仿真结果如图 9 所示, 分别给出了三轴正常对地下的地惯坐标系、地固坐标系和卫星本体坐标下二维和三维视角下的状态演化过程。这种星地同步高保真数字化仿真平台的现实意义在于: 一旦卫星出现故障, 可马上通过该数字化平台进行在轨故障快速复现, 进而开展后续故障隔离和修复, 提高卫星在轨寿命。

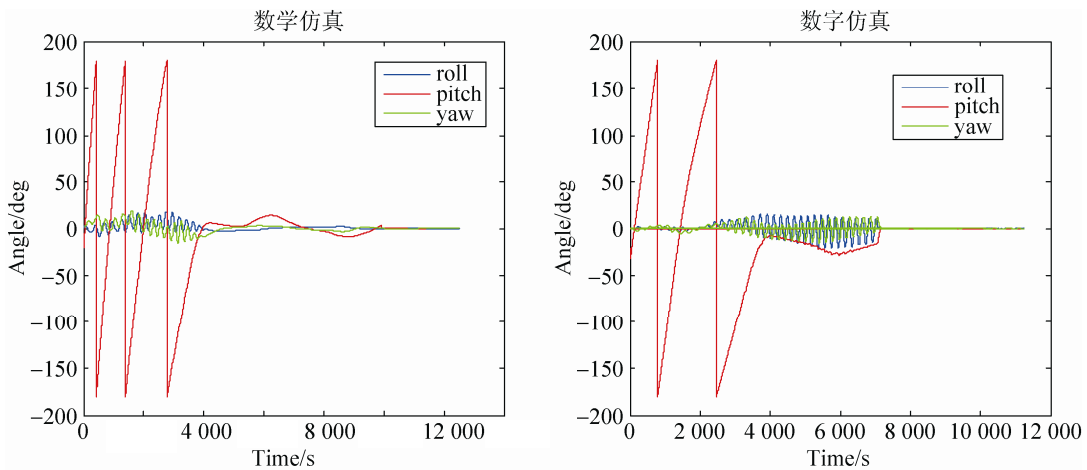


图 8 数学 vs 数字仿真

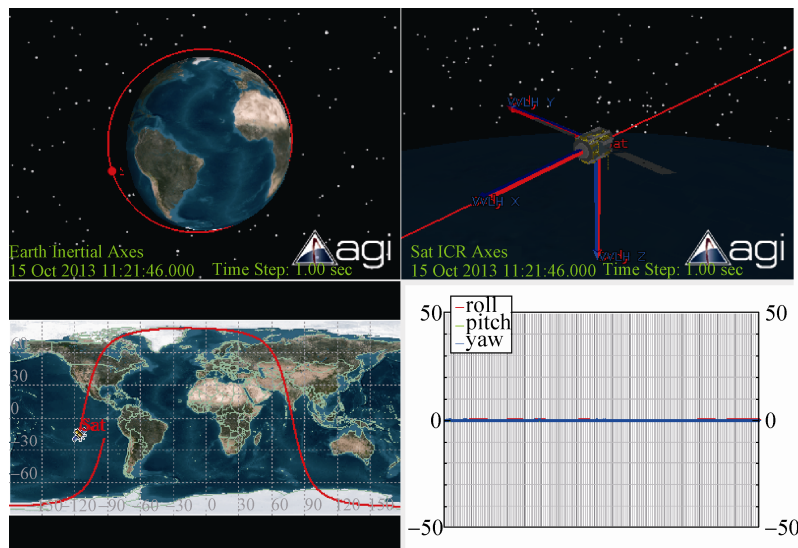


图 9 与 STK 联动高保真仿真示意图

<http://www.china-simulation.com>

5 结论

以实物 CPU 或目标板为核心的半物理测试环境很难建立特殊的仿真场景,特别是对于运行在动态 I/O 环境中的航天嵌入式软件,发生总线流量限制或丢包在内的各种硬件错误,会导致相关在轨故障难以快速复现。同时,在硬件测试环境中可能不支持为定位软件问题而进行的特定设置,例如,触发伪中断,单粒子翻转等内存故障,造成软件运行的“透明性”非常有限。因此,一旦卫星发生在轨故障将会以极大的代价通过故障发生前的遥测数据进行场景复现,而延缓了后续故障定位及排除。

与此相对应,数字虚拟化环境却提供了硬件环境所不具备的能力,例如触发伪中断、修改 CPU 寄存器内容、在通讯接口层模拟外设故障等,这些能力对于复现和定位特定的系统故障是非常必要的。航天器一旦发生故障,通过将数字化虚拟平台的虚拟硬件和软件状态设置为故障发生时的状态,然后按照时间逆向推演,观察和查找引起故障前关键状态变量的变化情况,经过数据分析可快速定位出故障发生诱因,大大提高在轨维护的效率。

相比于传统半物理仿真平台,数字化平台在故障复现、定位等在轨维护方面具有如下优势:

- 1) 时间尺度可调,并且可回溯;
- 2) 从故障发生时间点可逆向运行,其过程中可设置断点和观察点;
- 3) 在一次运行中可精确观察数据流内部状态,做到全程监控;
- 4) 故障类型丰富多样,且实现代价极低

基于 CPU 模拟器的虚拟环境在航天器在轨维护和故障复现中已得到了实际应用,尤其是在多颗低轨卫星故障定位中起到了决定性作用。

由于在仿真中软件的执行过程高度可见,并完全受控,系统的状态也完全受到监控。相较于传统方法全数字仿真技术对于故障定位而言有着无可比拟的优势,使得卫星在轨维护获得了一种更为经济且有效的技术手段。因此,在卫星在轨维护方面应尽可能多地发挥虚拟仿真环境的作用,为星载软

件执行提供系统级、高真实性的运行、仿真和测试的综合支撑平台。

全数字仿真技术是一种全新的系统集成、仿真验证模式,势必会带来对航天领域系统仿真相关概念与技术的重大变革。当然,数字化虚拟平台并不能完全替代硬件平台,却是对硬件平台的重要而有效的补充。基于虚拟平台进行航天嵌入式系统集成、软件开发和测试的优点也将会日益被人们认识,应用范围也会越来越广。

参考文献:

- [1] 陈小前,袁建平,姚雯. 航天器在轨服务技术 [M]. 北京: 中国宇航出版社, 2009.
- [2] 杨海峰,张发家,刘吉霞,等. 卫星在轨故障全数字仿真平台设计 [J]. 航天器工程, 2012, 21(4): 21-25.
- [3] Jens Eickhoff. Simulating Spacecraft Systems [M]. Berlin, Germany: Springer-Verlag Press, 2009.
- [4] 刘良栋. 卫星控制系统仿真技术 [M]. 1 版. 北京: 宇航出版社, 2003.
- [5] James E Smith. Virtual Machines Versatile Platforms for Systems and Processes [M]. USA: Morgan Kaufman, 1999.
- [6] 周玉宇,尚利宏,吕紫旭. 基于虚拟机的单粒子翻转故障注入方法 [J]. 计算机研究与发展, 2010, 47: 310-313.
- [7] 郭华,武占峰,吴瑾. 全数字仿真测试环境在星载嵌入式软件测试中的应用 [J]. 航天控制, 2012, 30(6): 70-72.
- [8] SPARC 7 Instruction Set [Z]. Germany: SPARC International, Inc, 2000.
- [9] The SPARC Architecture Manual v7 [Z]. Germany: SPARC International, Inc, 2000.
- [10] Embedded Processor 32-bit SPARC User's Manual [K]. Germany: SPARC International, Inc, 2000.
- [11] 熊淑杰,张锐,白涛. 卫星姿轨控半物理仿真试验系统构架研究 [J]. 系统仿真学报, 2014, 26(7): 1487-1491.
- [12] 屠善澄. 卫星姿态动力学与控制 [M]. 1 版. 北京: 中国宇航出版社, 1998.
- [13] 门斯布吕克,吉尔. 卫星轨道: 模型、方法和应用 [M]. 1 版. 北京: 国防工业出版社, 2012.
- [14] Petit G, Luzum B, Al E. IERS Conventions (2010) [Z]. McCarthy, USA: IERS Conventions Centre, 2010.
- [15] 蓝枫叶. 自己动手写嵌入式操作系统 [M]. 2 版. 北京: 电子工业出版社, 2009.
- [16] 西迪. 航天器动力学与控制 [M]. 1 版. 北京: 航空工业出版社, 2011.
- [17] 丁溯泉,张波,刘世勇. STK 在航天任务仿真分析中的应用 [M]. 1 版. 北京: 国防工业出版社, 2011.