

基于软件和逻辑联合仿真的 SOPC 验证技术研究

周 珊,王金波

(中国科学院 空间应用工程与技术中心,北京 100094)

摘 要: 针对航天高可靠 SOPC 测试验证重要性,对 SOPC 仿真技术进行深入剖析研究,提炼出了一套切实可行的 SOPC 仿真技术. 阐明了 SOPC 仿真技术的具体实现方法,分析了 SOPC 的具体验证方法,并把这套仿真验证技术成功地应用到了多个航天高可靠 SOPC 的测试中,发现了很多引起功能失效的重大问题,并对其中常见的问题给予归类总结.

关键词: SOPC 测试验证;SOPC 仿真技术;CPU 内核验证;仿真环境;软件和逻辑联合验证

中图分类号: TP391.9

文献标识码: A

文章编号: 1000—7180(2016)06—0082—05

SOPC Verification Technology Research Based on Software and Logical Combined Simulation

ZHOU Shan, WANG Jin-bo

(Technology and Engineering Center for Space Utilization, Chinese Academy of Sciences, Beijing 100094, China)

Abstract: For the significance of high reliable SOPC test verification on Spaceflight application, SOPC simulation technology is analyzed and researched, and a set of feasible SOPC simulation technology is extracted. The specific realization method of SOPC simulation technology is clarified. The specific verification method of SOPC is analyzed, the test verification method between SOPC software and logical combined test. And the simulation verification technology is applied successfully on several reliable SOPC tests, many significant functional disabled problems can be found, and the common problems are classified and summarized.

Key words: SOPC test verification; SOPC simulation technology; CPU kernel verification; simulation environment; test verification method between SOPC software and logical combined test

1 引言

SOPC(可编程片上系统)技术是 SOC(片上系统)和 FPGA(可编程逻辑)技术结合的产物,它是将处理器、存储器、I/O 接口、422 协议接口等系统设计需要的元素集成在一个可编程器件中,具备可裁剪、可升级、可扩充、软硬件可编程等优点^[1-2]. 基于上述特点, SOPC 在空间应用环境中的运用越发广泛,随着 SOPC 的广泛应用,功能日趋强大,其系统的设计也日趋复杂. 当 SOPC 系统的规模越来越大时,其可靠性变得越来越突出,由于编程规范、IP 异常使用、软硬件交互等引发的

功能错误在空间环境中时有发生,因此对 SOPC 的测试需求变得尤为迫切. 针对于 SOPC 的验证技术在业界还没有统一的标准,所以有必要研究一套切实可行且经过大量工程检验的验证方法.

目前基于 SOPC 的测试验证方法主要是基于以下两个方向的研究:一是基于黑盒的板级动态测试方法^[3-4],二是基于白盒分析的仿真技术.

以上两种方法都有一定的局限性,本文提炼了一套基于软件逻辑联合仿真的 SOPC 验证技术,并将这一技术应用在了多个型号任务的 SOPC 测试中,发现了很多能引起功能失效的重大问题.

2 软件和逻辑联合仿真的 SOPC 验证技术

基于软件逻辑联合仿真的 SOPC 仿真技术具备如下优势:

- (1)软件和逻辑都基于原始代码,避免了用逻辑代码模拟软件引起的功能或时序失真;
- (2)软件的线程调度、优先级判定、中断响应、进程执行顺序、进程执行时间、变量的值都可以通过波形直观显示,便于定位问题以及潜在风险;
- (3)逻辑代码和软件代码的交互接口通过波形查看器可以直接查看,便于查找交互错;
- (4)网表层面的仿真能反映真实器件的时序,便于分析时序路径的时序是否满足要求。

本文以 xilinx 的 FPGA 芯片实现的 SOPC 系统来具体论述基于软件和逻辑的联合仿真技术, CPU 内核采用 PPC405 硬核,仿真工具采用 mentor 公司的 questasim. 主要论述与传统纯逻辑代码仿真不同之处,图 1 所示包括前仿真和仿真库的配置、仿真软件初始化文件配置、测试 TB 配置、内核调试方法、仿真结果分析方法等。

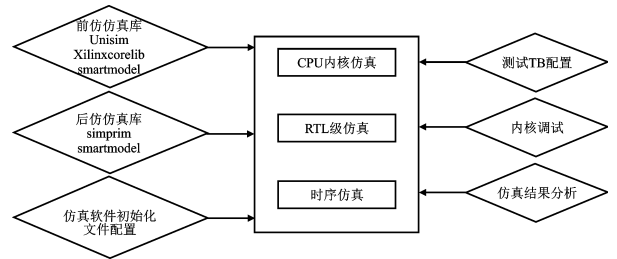


图 1 SOPC 仿真技术流程

2.1 仿真库

仿真工具对于代码或网表的仿真实际上是把代码或网表转换成底层库单元中的器件连接,通过库单元中器件的行为模型来模拟整个代码或网表的行为^[5-7]. 基于逻辑代码仿真时,只需把 FPGA 厂商提供的库器件源文件按实际使用的仿真工具编译后即可使用. 基于 SOPC 的仿真时,SOPC 对应的 CPU 内核以及 CPU 内核周边连接的其他软核,如总线控制器、数据存储器和程序存储器、GPIO 核等在仿真工具中也将转换为底层库单元中的模型,所以库单元中须有这些软核对应的行为模型. SOPC 仿真时除了用到和纯逻辑仿真时的库如 UNISIM 库、UniMacro 库、XilinxCorelib 库、SIMPRIM 库等,还需用到如下库单元.

(1)EDK 仿真库

EDK 库是用于 SOPC 行为级的仿真库,它包括所有 EDK IP,如 xps_uart、ppc_405、sdram 控制器等等,SOPC 在前仿真时必须调用 EDK 的库,EDK IP 才能表现出行为动作.

(2)SmartModel/SecureIP 仿真库

SmartModel/SecureIP 用于复杂 FPGA 单元器件的行为和时序仿真的验证模型,如 Virtex-4 器件和 Virtex-5 器件包括了 RockeIO、PowerPC 处理器,以及 Ethernet MAC 等硬核. Smartmodel 是用实际硬件描述语言(HDL)描述的加密版本的硬核,它给出了硬核的行为表现. 如 PPC_405 在 EDK 库中表现形式为 PPC_405 的外部接口封装加上内部 PPC 核的调用,内部 PPC 的行为表现体现在 SmartModel/SecureIP 仿真库中. 仿真时 SmartModel/SecureIP 库和 EDK 仿真库必须联合使用.

2.2 仿真工具初始化文件修改

modelsim. ini 文件是运行仿真工具 Questa 时的初始化文件,在进行 SOPC 的仿真时,需做如下的更改.

(1)添加 ISE 库文件:编译库在初始化文件中设置路径.

(2)环境变量设置:

```
Libsm= $ model_tech/libsm. dll  
Libswift= $ LMC_HOME/lib/pcnt. lib/swift. dll
```

2.3 仿真环境建立

仿真环境是被测代码或网表周边的虚拟验证环境,虚拟验证环境和实际环境相比具有测试激励实现简单,施加的时刻精准,施加的对象可以是接口或是内部信号,可以模拟任意功能和接口场景,定位问题方便等优点,所以仿真测试是可编程逻辑验证的重要环节. SOPC 仿真和逻辑仿真在后仿真层面仿真环境无差异,在前仿真时因为 PPC 的初始化文件单独生成,所以需要把初始化文件加载到 PPC 中. 具体如下,在 TB 文件中添加以下代码 [7]:

```
configurationTB_TOP_conf of TB_TOP is  
for STRUCTURE for all :顶层文件  
use configuration work. 配置文件名  
endfor; endfor; endTB_TOP_conf;
```

2.4 内核调试

内核仿真分析是确认 PPC 是否真正运行起来,调试的方法为给予 PPC 接口有效地时钟信号,复位信号后查看 PPC 输出接口的信号是否按照软件代码运行的顺序进行初始化的工作. 如果为带操作系

统的 PPC 仿真则首先验证所有线程是否正确创建,并且创建完的线程是否按优先级调度的机制在运行. 以下为调试带操作系统的 PPC 步骤.

操作系统:ucose; cpu 核:ppc_405; 仿真环境:QuestaSim 6. 5d; 工程环境:ISE 10. 1. 03;操作系统运行机制:优先级调度的方式.

(1)确认线程是否创建完成

在 PPC 主程序中增加调试信息,便于确认线程是否创建,如下所示.

```
Main {Init_System();
WriteGPIO(XPAR_GPIO_CTR_BASEADDR, 0x01,
0x00000001);//调试信息
OSInit();
WriteGPIO(XPAR_GPIO_CTR_BASEADDR, 0x01,
0x00000002);//调试信息
OSTaskCreate(TaskStart, NULL,&g_StkStart[ SIZE-
1 ],TASK_PRIOR_START );/*
```

```
WriteGPIO(XPAR_GPIO_CTR_BASEADDR, 0x01,
0x00000003);//调试信息
OSStart(); /* /.....}
voidTaskStart( void * pData ) {
OSTaskCreate(TaskRT, NULL,
&g_StkRT[ SIZE-1 ], TASK_PRIOR_RT );
WriteGPIO(XPAR_GPIO_CTR_BASEADDR, 0x01,
0x10000001);//调试信息
OSTaskCreate( TaskRepair, NULL, &g_StkRepair[
SIZE-1 ],
TASK_PRIOR_REPAIR )
WriteGPIO(XPAR_GPIO_CTR_BASEADDR, 0x01,
0x30000001)//调试信息
OSTaskCreate( TaskDown, NULL, &g_StkDown [
SIZE-1 ],TASK_PRIOR_DOWN );
WriteGPIO(XPAR_GPIO_CTR_BASEADDR, 0x01,
0x50000001);
```

仿真结果如图 2 所示.

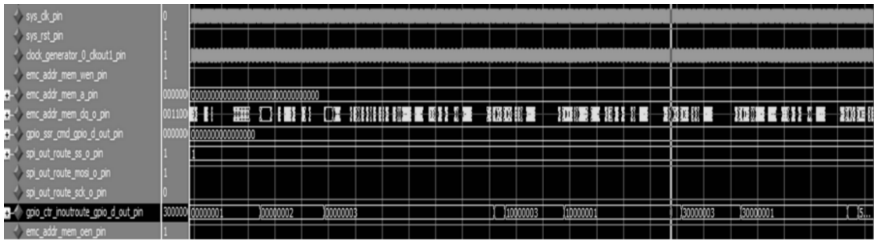


图 2 线程创建波形

仿真结果显示:PPC 的软件代码已成功运行,在主程序中按照顺序首先进行了应用程序的初始化操作(00000001)以及操作系统的初始化操作(00000002);然后进行了主线程的创建(00000003);在主线程中创建了 3 个子线程(10000001, 300000001,500000001).

(2)确认各个线程的调度方式为优先级调度

代码设计时线程 TASK_PRIOR_RT 的优先级最高,线程 TASK_PRIOR_REPAIR 的优先级为中,线程 TASK_PRIOR_DOWN 的优先级最低. 在每个线程的开始时加入如下调试信息,其余两个线

程加入类似的调试信息:

```
voidTaskRT( void * pData )
{WriteGPIO(XPAR_GPIO_CTR_BASEADDR, 0x01,
0x10000002);//调试信息
while(1) {WriteGPIO(XPAR_GPIO_CTR_BASEAD-
DR, 0x01, 0x10000003);//调试信息
OSTimeDly(30);}}
```

通过仿真结果可以确定线程执行的顺序,实际运行的仿真如图 3 所示. 代码在实际运行时候是按照实际优先级的高低来执行,并且可以精确测量每个线程实际运行的时间.

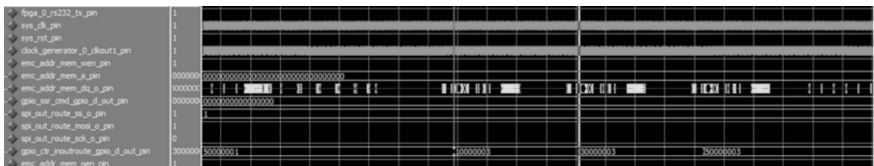


图 3 线程优先级调度波形

(3)确认 PPC 的接口信号是否有效

PPC 的对外接口是和逻辑代码交互的通道,接

口数据流的交互很大程度影响 SOPC.

功能的正确性. 所以在功能仿真之前要确认接

口交互信号的正确性,PPC 的接口有总线接口,数据 RAM 接口,程序 RAM 接口,JTAG 链接口等.与内部逻辑交互的接口主要包括 GPIO 接口,BRAM 接口等. GPIO 接口通常利用其中的某位或

多位传递控制信号或数据信号;BRAM 接口是传递大量数据和状态信息的缓冲区,通过这个缓冲区可以把发送和接收大数据信息传递到下一级. PPC 接口的仿真图,如图 4 所示.

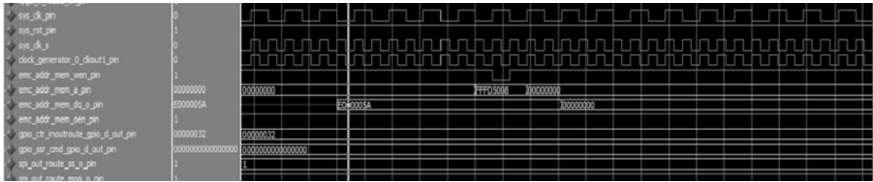


图 4 PPC 接口仿真波形

由图 4 可以看出,PPC 对外输出了 CLK,通过 mem 向某个逻辑地址写入了数据,所以可以证明 PPC 对逻辑的接口是能进行数据交互的.

3 仿真结果分析

CPU 内核仿真起来后,就可以联合逻辑代码进行功能和时序仿真. 功能仿真主要验证功能需求是否正确;时序仿真首先验证内部路径的建立保持时间以及恢复重聚时间是否满足要求,其次验证外部接口以及 PPC 接口时序是否正确.

(1)功能仿真

功能仿真测试软件、逻辑,以及联合云行的正确性,是代码行为方式的一种直观表现,一定程度上可以反映代码的功能正确性. 本文以 SOPC 实现 1553B 总线芯片的主控功能为例来说明功能的仿真验证,其中 PPC 软件代码实现业务流的控制,逻辑

代码实现功能到 1553B 协议层的时序转换.

①根据内核和逻辑交互的信号确认软件代.

一般嵌入式软件都是通过调试器查看寄存器变量值来进行调试验证,通过实测结果来确认功能正确性. SOPC 中软件代码是通过仿真验证方式,查看软件对外接口信号波形来调试验证,内部变量可以通过赋值给外部接口验证其值的正确性. SOPC 的仿真打破了传统意义上软件代码的验证方式,可以更加直观、清晰地看到软件执行顺序、线程的优先级、进程执行的时间,以及各个函数之间交互的信号量变化等.

由图 5 可以看出,PPC 软件在以一定的时间间隔向不同的地址发送相应的数据. 首先是向 1553B 的寄存器地址里写入数据,接着向 1553B 的存储器地址里写入数据,配置数据的正确性可以参照业务层协议和数据链路层面的协议进行验证.

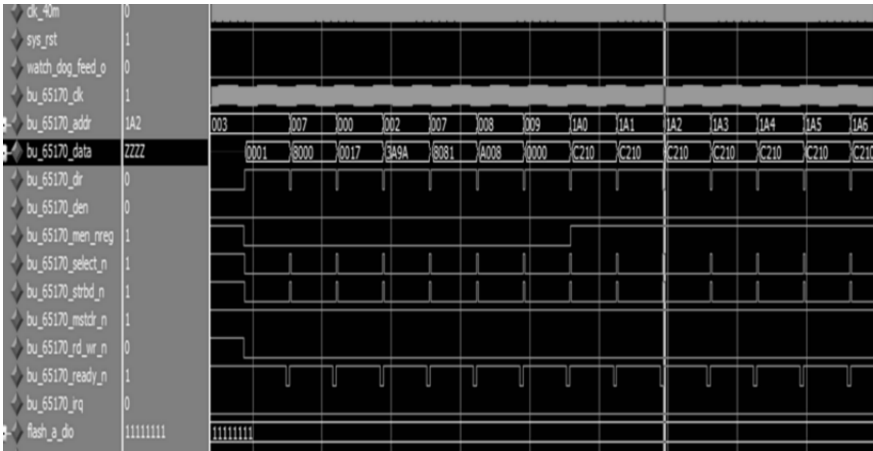


图 5 功能仿真波形

②根据内核和逻辑代码的运行表现确认功能是否正确,并且可以通过内部信号调试来分析代码的风险和错误的机理.

软件功能验证正确的基础上可以结合逻辑代码

进行 SOPC 行为级的验证,本例中的逻辑代码主要实现 1553B 信号协议层的转化. 由图 5 看出 1553B 控制信号包括读写信号(rd_wr_n),片选信号(select_n),寄存器存储器选择信号(mem_nreg)等,这些控

制信号在波形中表现了相应的时序关系,通过 1553B 芯片手册确认其时序正确性。

(2)时序仿真

SOPC 时序仿真与逻辑时序仿真差别在于, SOPC 逻辑代码的接口包含两部分:一部分为外部接口,另外一部分为 PPC 的接口. 通过 SOPC 的时序仿真可以验证以下两方面.

①PPC 接口信号时序

验证 PPC 接口与内部逻辑交互时候是否存在时序不满足情况或者亚稳态情况.

②时序仿真引起的功能表现是否正确

验证在实际网表层面典型功能场景是否正确,在行为级验证中已验证过行为表现的正确性,在网表层面如果验证不通过,可以通过波形追溯出哪些信号时序的不满足引起了功能层面的错误.

4 SOPC 仿真技术的应用

上述的 SOPC 仿真技术应用在了航天高可靠型号任务 SOPC 片上系统测试中,共测试验证了 5 个项目,发现了以下几类问题.

(1)软件代码的线程优先级调度出现问题.

(2)软件代码对于 GPIO 的某一位操作时,误操作了其他位,导致给逻辑代码发出的控制信号混乱,引起功能错误.

(3)软件代码的进程安排不合理导致控制流信号混乱引发功能异常.

(4)软件代码在使用需外部接口触发的中断时,由于中断处理方式不正确引发功能异常.

(5)逻辑接口和 PPC 接口交互时由时序问题引起建立保持时间不满足导致功能异常.

(6)逻辑采集数据存放在存储区,软件没有及时取走数据,导致数据上传出现故障.

测试发现,上述几类问题引发了许多重要功能的不满足,在传统板级测试或模拟软件功能的纯逻辑测试验证中很难定位问题的原因,基于软件和逻辑

联合仿真的 SOPC 验证技术则可以准确定位出问题,避免引发功能异常或风险.

5 结束语

本文对 SOPC 仿真技术进行深入剖析研究,提炼出了一套切实可行的基于软件和逻辑联合仿真的 SOPC 验证方法. 阐明了 SOPC 仿真技术的具体实现方法,包括软件工具初始化文件的配置、处理器仿真库的配置、仿真环境的特殊配置. 分析了 SOPC 的具体验证方法,包括 CPU 内核测试方法、内核对逻辑代码接口测试方法、SOPC 软件功能测试方法、SOPC 软件和逻辑联合测试验证的方法. 并对其中常见的问题给予归类总结. 这套验证方法可以作为 SOPC 仿真验证的一个标准,在工程实践中存有重要价值.

参考文献:

- [1] 陈叶富. 集成处理器仿真平台 SOPC 系统可测试性方法研究[D]. 哈尔滨:哈尔滨工业大学,2011.
- [2] 姚双武. 基于 SOPC 的集成处理器仿真平台研制[D]. 哈尔滨:哈尔滨工业大学,2010.
- [3] 刘景宁,舒芳,童薇,等. SOPC 光纤通道控制器 IP 核的仿真验证[J]. 华中科技大学学报,2008,36(10): 91-94.
- [4] 沈浩. 基于 SOPC 的短波 OFDM 通信信号仿真平台的研究[D]. 武汉:武汉工程大学,2011.
- [5] 魏子杰,许军,张磊,等. 基于 SOPCde 电力电子实时仿真系统设计[J]. 电测与仪表,2012 年,49(11):81-84.
- [6] 姚凤振. 基于 SOPC 技术的水声通信半物理仿真平台设计与实现[D]. 青岛:中国海洋大学,2013.
- [7] xilinx ISE help[EB/OL]. [2015-08-01]. <http://help.ubuntu.com/communtly/xilinxISE>. 2014.

作者简介:

周 珊 女,(1983-),硕士,工程师. 研究方向为高可靠软件,可编程逻辑测试. E-mail:zhoushan@csu.ac.cn.

王金波 男,(1978-),博士,副研究员. 研究方向为高可靠软件.

(上接第 81 页)

[8] 王秦. 基于 OTP 的移动商务身份认证协议研究[D]. 北京:北京交通大学,2010.

[9] 叶汝军. 校园网络身份认证技术平台研究[J]. 微电子学与计算机,2012,,29(3):155-158.

作者简介:

王增光 男,(1991-),硕士研究生. 研究方向为网络安全与信息对抗. E-mail:wang1223797579@163.com.