

doi:10.13756/j.gtxyj.2024.230104.

光电器件研究与应用

李文艺,徐志生. 面向芯片驱动的 SDK 仿真平台研究与实现[J]. 光通信研究, 2024(6):230104.

Li W Y, Xu Z S. Research and Implementation of SDK Simulation Platform for Chip Drivers [J]. Study on Optical Communications, 2024(6):230104.

# 面向芯片驱动的 SDK 仿真平台研究与实现

李文艺<sup>1</sup>, 徐志生<sup>2</sup>

(1. 武汉邮电科学研究院, 武汉 430074; 2. 烽火通信科技股份有限公司, 武汉 430074)

**摘要:**【目的】随着集成电路(IC)设计与制造工艺的不断升级,片上系统(SoC)的设计规模越来越庞大,相对应的软件开发工具包(SDK)也越来越复杂,缩短 SDK 的开发时间和快速应用是芯片研发成功的关键环节。【方法】文章设计了一种芯片 SDK 仿真平台,该平台支持对芯片逻辑设计信息进行自动化提取,基于设计的芯片数据结构实现对提取到的逻辑信息进行软件本地镜像、开发配置、告警和性能多线程来模拟实际单盘应用时的多任务轮询上报网管,通过虚拟串口连接 SDK 工程与串口工具,Shell 命令集提供 SDK 仿真平台下开发人员与 SDK 工程的交互。【结果】基于光通信芯片的测试结果表明,该 SDK 仿真平台可实现脱离硬件环境下的芯片软件级仿真、多线程调试模拟和 Shell 命令集下的多功能交互等功能。【结论】文章研究与实现的 SDK 仿真平台可应用于芯片前期开发与后期应用维护,避免了硬件资源紧张对开发的阻塞,收敛了芯片转产的差异,提供了便利的 SDK 调测方法,对缩短研发周期和加速芯片快速应用有着重要意义。

**关键词:**片上系统;软件开发工具包;芯片驱动;光通信芯片

**中图分类号:**TN914.3

**文献标志码:**A

## Research and Implementation of SDK Simulation Platform for Chip Drivers

LI Wenyi<sup>1</sup>, XU Zhisheng<sup>2</sup>

(1. Wuhan Research Institute of Posts and Telecommunications, Wuhan 430074, China;

2. FiberHome Telecommunication Technologies Co., Ltd., Wuhan 430074, China)

**Abstract:** 【Objective】With the continuous advancement of Integrated Circuit (IC) design and manufacturing process, the design scale of System on Chip (SoC) has become increasingly larger, and the corresponding Software Development Kit (SDK) has also become more complex. Shortening the development time and rapid application of SDK is the key to the success of chip development. 【Methods】In this paper, a chip SDK simulation platform is designed, which supports automatic extraction of chip logic design information. The software local image of extracted logic information based on the designed chip data structure is realized. The multi-threading of configuration, alarm and performance is developed to simulate the multi-task polling and network management reporting in the actual single-disk application. By connecting the SDK project to the serial port tool through the virtual serial port, the Shell command set provides interaction among developers and the SDK projects under the SDK simulation platform. 【Results】The test results based on optical communication chip show that the SDK simulation platform can realize the functions of chip software level simulation, multi-thread debugging simulation and multi-function interaction under Shell command set. 【Conclusion】The SDK simulation platform implemented in this paper can be applied to the early development and late application maintenance of chips. It has the functions of avoiding the development obstruction caused by the shortage of hardware resources, converging the difference of chip conversion, and providing a convenient SDK debugging method, which is of great significance to shorten the development time and accelerate the rapid application of chips.

**Key words:** SoC; SDK; chip driver; optical communication chip

## 0 引 言

随着“中国制造 2025”发展战略的全面实施,集成电路(Integrated Circuit, IC)产业蓬勃发展<sup>[1-3]</sup>。知识产权(Intellectual Property, IP)核是 IC 产业的支撑基础,其重复应用可以减少芯片设计的工作量,缩短开发周期,提高芯片设计的成功率<sup>[4-5]</sup>。

复杂芯片设计和 IP 核复用的软件开发工具包

(Software Development Kit, SDK)的开发面临着在较长开发周期下,前期硬件资源紧张给 SDK 功能调试带来的困难<sup>[6-7]</sup>,同时要解决 IP 核复用背景下,SDK 在多芯片间共用代码的适配及多芯片联合调试的问题<sup>[8]</sup>。

基于上述背景,本文研究并实现了一种面向芯片驱动的 SDK 仿真平台,该平台支持在脱离硬件的环境下对芯片 SDK 进行调试与验证,确保 SDK 的

收稿日期:2023-09-11; 修回日期:2023-11-05; 纸质出版日期:2024-12-10

作者简介:李文艺(2000—),女,湖北武汉人。硕士,主要研究方向为电子与通信技术、光通信技术。

通信作者:徐志生,工程师。E-mail:zsxu0807@fiberhome.com

© Editorial Office of *Study on Optical Communications*. This is an open access article under the CC BY-NC-ND license.

正确性,同时适配多芯片同步开发与维护,提高了代码的重用性,减少了后续开发的工作量。

1 SDK 仿真平台介绍

SDK 是为特定的软件包、软件框架、硬件平台和操作系统等建立应用软件时开发工具的集合<sup>[9]</sup>。本文提出了一种基于光通信芯片的 SDK 仿真平台,可支持芯片硬件信息自动化提取并镜像到本地软件内存、多线程支持模块模拟转产后的多线程调度和 Shell 模块实现用户与 SDK 仿真工程的交互等功能。

为了模拟验证板上的登录和调试过程,本文设计了虚拟 Uart 通信模块,利用串口通信完成登录工具与 SDK 仿真平台之间数据和命令的双向转发,总体设计如图 1 所示。

自动化芯片信息提取模块可根据芯片逻辑设计自动提取寄存器信息,形成特定格式的寄存器描述文件。

芯片软件模块是与芯片逻辑设计同步开发的芯片软件集,其中包括芯片驱动开发设计的芯片驱动软件集、调试功能集和算法功能集等。

Shell 模块支持芯片对外应用级接口自动生成对应的 Shell 命令集,同时还包括软件中所有的调试命令集、算法功能集和对外接口等。

多线程支持模块实现多线程支持功能。在仿真过程中通常启动5个线程:1个模拟系统主线程,

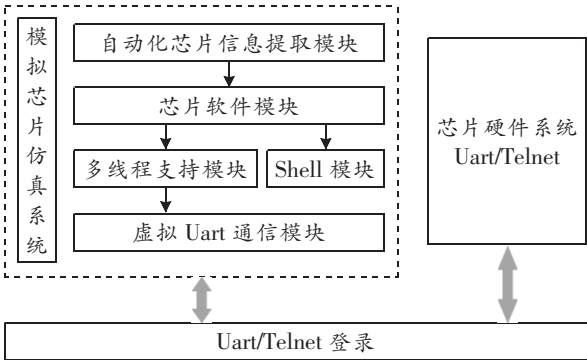


图 1 SDK 仿真平台总体设计

Figure 1 Overall design of SDK simulation platform

3 个分别模拟芯片应用的配置线程、告警上报线程和性能采集线程,1 个 shell 线程。

2 SDK 仿真平台的功能设计

2.1 芯片数据镜像流程

为了实现芯片硬件信息镜像到软件环境,并且满足多芯片共同开发与维护的需求,本文设计了如图 2 所示的芯片数据镜像流程。启动配置线程后调用芯片创建函数,创建对应芯片 SDK 并记录信息,完成后调用芯片初始化加载默认值。多芯片需求时重复上述流程,并从已完成镜像的芯片中整体拷贝信息。其中创建芯片 SDK 的具体流程如下:

① 获取芯片信息

将芯片类指针 `sys_info` 指向本地定义记录芯片

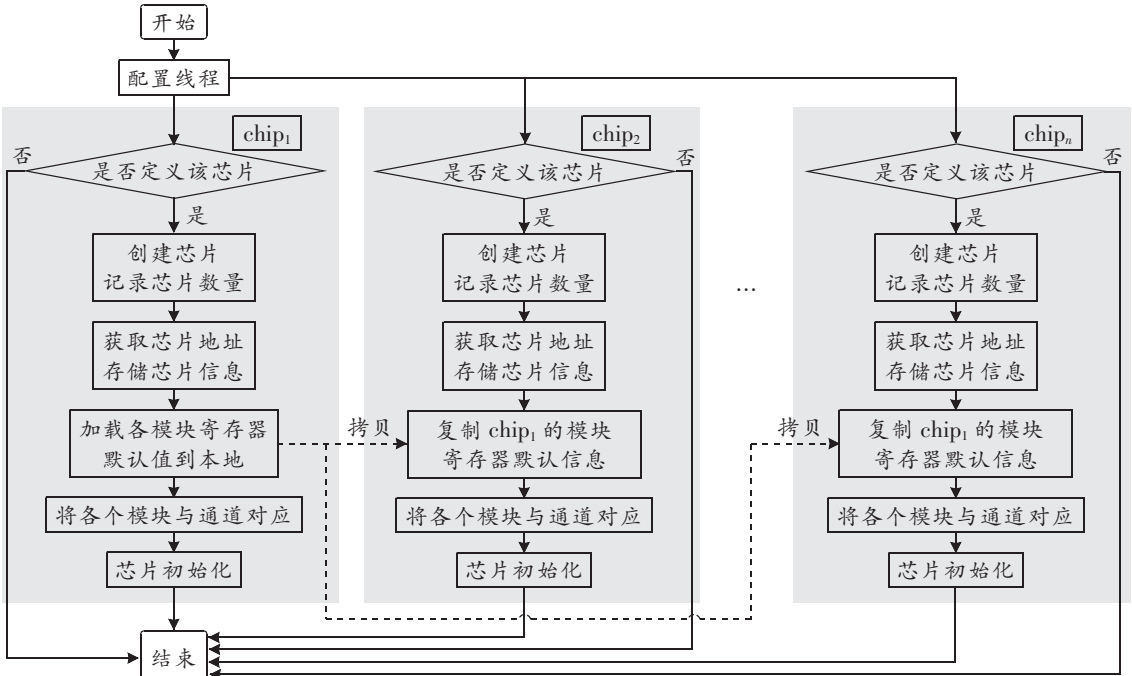


图 2 芯片数据镜像流程

Figure 2 Chips data mirroring process

信息的类中拷贝芯片信息。

② 存储芯片信息

*chip\_addr* 记录芯片地址,以芯片号为索引存储 *sys\_info* 和模块实例数量并动态分配内存空间。

③ 加载芯片信息

以模块为单位将自动生成的格式化寄存器信息加载到本地镜像环境中。其中,创建完成首片芯片 *chip<sub>1</sub>* 后,需继续创建同一芯片时(如 *chip<sub>2</sub>*、*chip<sub>3</sub>*、...、*chip<sub>n</sub>*),以模块为单位从 *chip<sub>1</sub>* 中整体拷贝加载芯片信息。

④ 模块与通道的对应

以模块为单位获取通道与模块的对应关系,根据模块序列号 *module\_id* 检索对应的模块地址 *module\_addr*。

2.2 芯片数据结构的设计

芯片随机存取存储器(Random Access Memory, RAM)依照逻辑功能进行物理块划分,再根据模块需求划分寄存器<sup>[10-11]</sup>。基于上述结构设计数据结构,在 SDK 内生成寄存器的镜像,从而模拟驱动接口对芯片寄存器的读写。

① *chip\_s* 为定义芯片类结构体。其中, *chip\_addr* 为芯片地址; *reg\_module\_num* 为芯片所有模块实例的数量; *module\_inst* 指向包含芯片模块实例信息的数组; *chip\_name* 为芯片名。

```
typedef struct chip_s{
    unsigned int * chip_addr;
    unsigned short reg_module_num;
    module_inst_s * module_inst;
    char * chip_name;};
```

② *module\_inst\_s* 为定义模块实例类结构体。其中, *module\_addr* 为模块实例地址; *module* 指向记录模块信息的模块类, *inst\_name* 为记录模块实例的名称。

```
typedef struct module_inst_s{
    unsigned int module_addr;
    module_s * module;
    char * inst_name;};
```

③ *module\_s* 为定义模块类结构体。其中, *reg\_num* 记录该模块内寄存器的数量; *regs* 指向记录寄存器信息的寄存器类; *module\_name* 为模块的名称,例化出的多套模块对应的模块名称相同。

```
typedef struct module_s{
    unsigned int reg_num;
    reg_info_s * regs;
```

```
char * module_name;};
```

④ *regs\_info\_s* 为定义寄存器类结构体。其中, *reg\_id* 为寄存器序列号; *offset\_addr* 为寄存器偏移地址; *default\_value* 为寄存器默认值; *value* 为本地模拟环境中的寄存器值; *reg\_bits* 指向记录寄存器各比特域高低位的寄存器比特类。

```
typedef struct reg_info_s{
    unsigned int reg_id;
    unsigned short offset_addr;
    unsigned int default_value;
    unsigned int value;
    reg_bit_info_s * reg_bits;};
```

2.3 多线程调试

为在芯片软件开发阶段模拟转产后的多线程应用,开发多线程模块模拟告警上报和性能提取来提前发现和规避芯片应用中可能存在的问题。

SDK 仿真工程运行时启动主线程模拟芯片硬件环境,对芯片地址和寄存器默认值等进行配置;并且创建 4 个子线程用于对 SDK 功能进行调试与验证。其中,配置线程模拟仿真工程与串口工具间的通信,对获取到的字符串命令进行解析并实现对应 SDK 接口功能;Shell 线程支持 Shell 命令的解析和实现;告警线程模拟告警上报;性能线程模拟性能提取。告警线程和性能线程在仿真环境下以固定的时间周期轮询,模拟对告警及性能信息的采集。

如图 3 所示,通过 4 对虚拟串口驱动程序(Virtual Serial Port Driver, VSPD)将 SDK 仿真工程与安全连接到远程终端(Secure Connection to Remote Terminal, Secure CRT)串口工具互联,实现命令的交互。开发人员在 COM11 可进行业务的配置,COM17 提供 Shell 交互功能,COM13 和 COM15 分别进行芯片告警监控和性能提取。

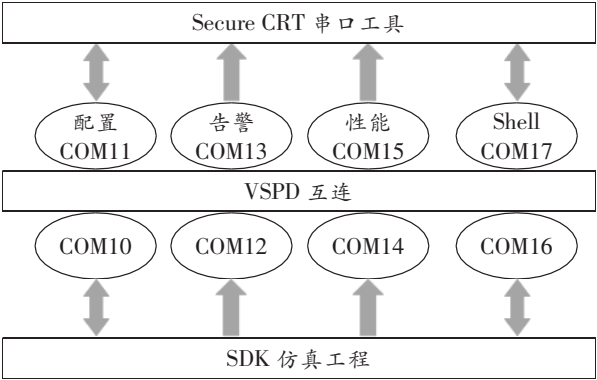


图 3 串口功能划分

Figure 3 Function division of serial port

2.4 Shell 命令集

为了便利仿真环境下的人机交互,设计 Shell 模块模拟 SDK 在硬件环境下的 Shell 命令集,实现开发人员对驱动接口、调试命令和算法功能集的便捷调用。Shell 作为一个独立的模块,能够应用于各芯片 SDK 的软件环境中,并且可以根据实际应用需求添加命令。

通过虚拟 COM16 与 COM17 的串口通信,在终端键入命令,SDK 仿真工程对命令进行解析。可支持多用户密码登录、历史命令条目配置及查询、查看和修改全局变量、按地址执行任意函数命令、SDK

接口函数调用和设置芯片号等功能。

3 SDK 仿真平台的功能验证

3.1 SDK 仿真平台搭建

SDK 仿真平台的搭建采用 Visual Studio 集成开发环境,Pthreads-w32 多线程库,VSPD 工具模拟主机与硬件环境的连接,Secure CRT 串口工具用于终端仿真。

如图 4 所示,当 SDK 仿真工程启动调试后,可通过终端查看各 VSPD,完成告警及性能信息的获取和开发人员与 SDK 仿真工程的交互。

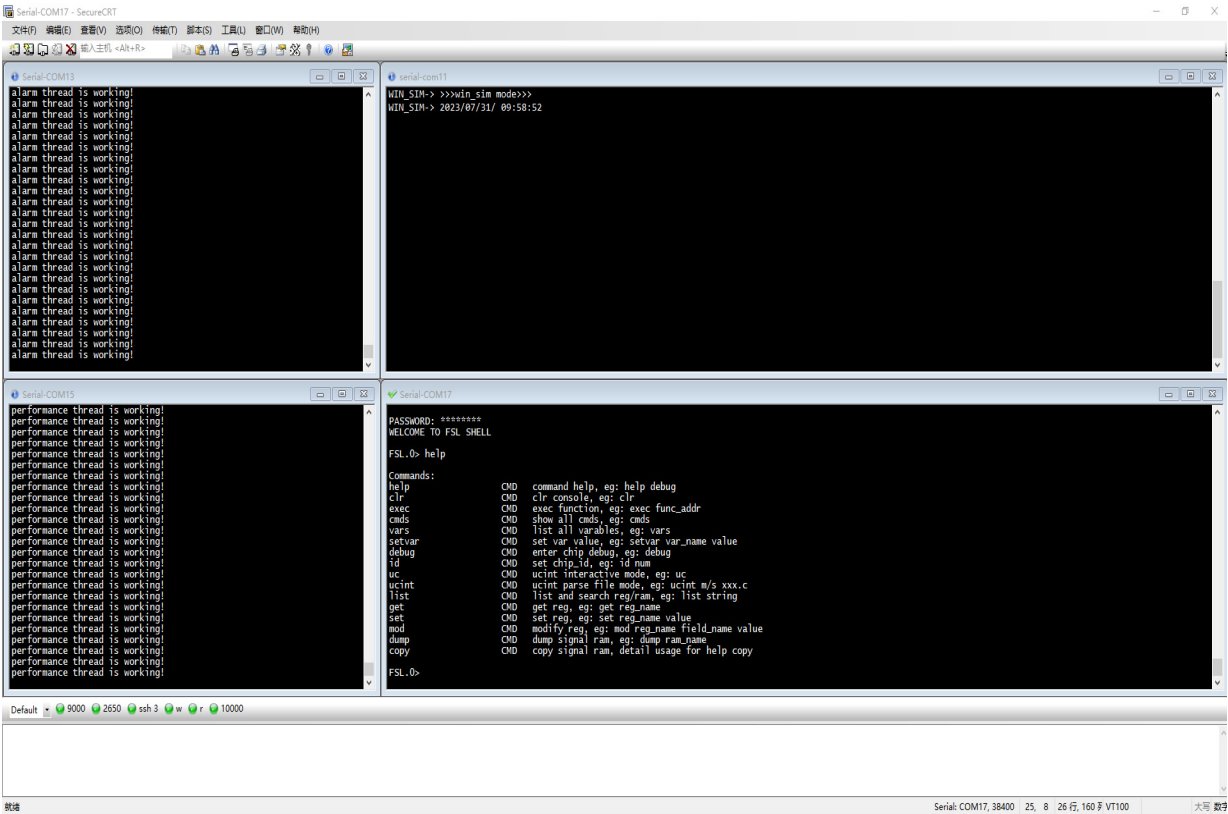


图 4 SDK 仿真平台搭建

Figure 4 Build SDK simulation platform

3.2 功能验证

图 5 所示为 SDK 仿真工程运行时,通过 COM13 获取告警线程模拟告警上报的结果。芯片告警线程和性能线程的轮询功能可以提前模拟芯片转产后的多线程应用,如网管对光通信芯片的告警和性能信息的获取。传统方案中对驱动的验证为现场可编程逻辑门阵列(Field Programmable Gate Array,FPGA)单任务运行,与转产应用后的多任务调度存在差异,一些隐蔽的任务调度问题无法在自测阶段暴露,而 SDK 仿真工程的多线程调用可以提前规避这些问题,提供完整的软件开发环境仿真。

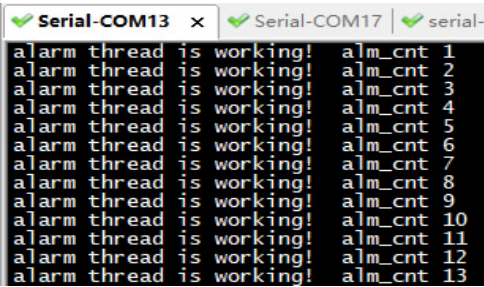


图 5 SDK 仿真平台对告警轮询的模拟

Figure 5 Simulation of alarm polling by SDK simulation platform

如图 6 所示,读取芯片寄存器,比较默认值,再

对该地址写入值 0x8,再次读寄存器,并比较读写结果。SDK 仿真平台对逻辑信息的本地镜像正确,且读写功能正常。传统方案中一些驱动问题依靠通用验证方法学(Universal Verification Methodology, UVM)仿真发现,无法定位中间过程,效率低的同时消耗仿真资源,并且这类问题随着软件算法的增加和复杂度的提高更加突出。SDK 仿真平台能够支持寄存器级的问题定位手段,发现问题后可在平台中复现,借助软件日志信息在集成开发环境中快速定位问题,提高驱动开发的效率,从而加速 UVM 仿真和逻辑设计工作。

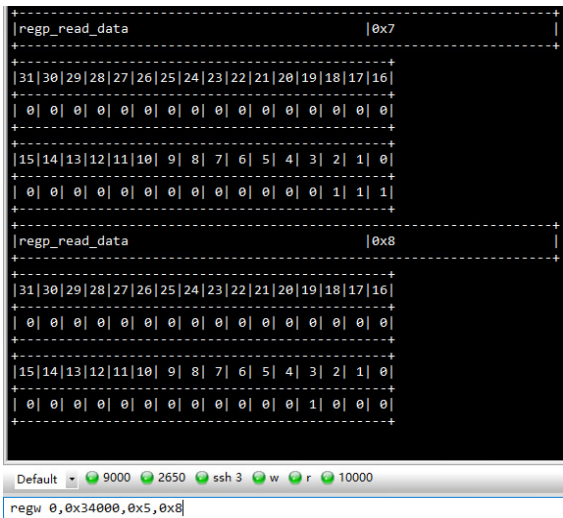


图 6 读写功能验证

Figure 6 Verify the read and write function

利用 Shell 命令集可以调用各个芯片对应的调试命令,如图 7 所示对光通路数据单元(Optical Da-

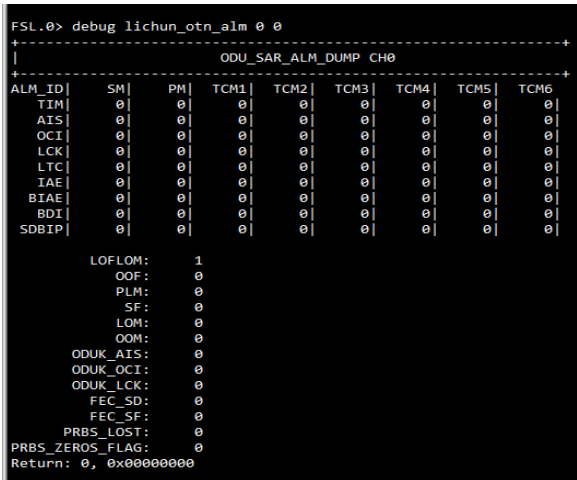


图 7 Shell 命令对芯片告警查询

Figure 7 Query chip alarms by Shell commands

ta Unit, ODU)告警查询发现上报帧丢失或复帧丢失(Loss of Frame and Loss of Multiframe, LO-

FLOM)告警, LOFLOM 告警在没有接收到任何 ODU 帧或复帧信号时触发。在后期应用中,网管发现告警信息后,可通过 Shell 命令集调用底层驱动接口定位具体告警,快速分析告警原因。

4 结束语

本文研究的 SDK 仿真平台设计了芯片逻辑数据镜像、多线程调试和 Shell 等功能,并基于大容量光通信芯片对上述功能进行了实现与应用。其实现优化了芯片 SDK 开发架构,可应用于芯片驱动开发、测试及维护的全过程,成功减少了芯片研发前期对验证板的依赖,降低了后期 SDK 调测和维护难度,对整体提升 SDK 开发效率、加速芯片转产应用和保证芯片的竞争力有着重要意义。

参考文献:

[1] 逯东, 池毅. 《中国制造 2025》与企业转型升级研究[J]. 产业经济研究, 2019(5):77-88.  
Lu D, Chi Y. Research on 《China Manufacturing 2025》 and Enterprise Transformation and Upgrading [J]. Industrial Economics Research, 2019(5):77-88.  
[2] 李先军, 刘建丽. 中国集成电路产业发展:“十三五”回顾与“十四五”展望[J]. 现代经济探讨, 2021(3): 87-96.  
Li X J, Liu J L. Development of Integrated Circuit Industry in China: Review of the 13th Five-year Plan and Prospect of the 14th Five-year Plan[J]. Modern Economic Research, 2021(3):87-96.  
[3] 张百尚, 商惠敏. 国内外芯片产业技术现状与趋势分析[J]. 科技管理研究, 2019, 39(17):131-134.  
Zhang B S, Shang H M. Current Situation and Trend Analysis of Chip Industry Technology at Home and Abroad[J]. Science and Technology Management Research, 2019, 39(17):131-134.  
[4] 刘伟平, 王宗源. EDA 产业与 IP 核产业发展现状及发展趋势[J]. 前瞻科技, 2022, 1(3):90-100.  
Liu W P, Wang Z Y. Development Status and Trend of EDA and IP Core Industries[J]. Science and Technology Foresight, 2022, 1(3):90-100.  
[5] 梁龙龙, 李钊, 郎少波. 系统工程中 IP 核的设计方法[J]. 电子技术与软件工程, 2021(19):79-80.  
Liang L L, Li Z, Lang S B. Design Method of IP Core in System Engineering[J]. Electronic Technology & Software Engineering, 2021(19):79-80.  
[6] 俞培勇. 基于 Groovy 的数据集成 GUI/SDK 自动化测试框架的研究与实现[D]. 上海:上海交通大学, 2015.  
Yu P Y. Development and Implementation of Data In-

tegration GUI/SDK Automation Testing Framework based on Groovy[D]. Shanghai, China: Shanghai Jiao Tong University, 2015.

[7] 段海霞. 基于系统总线的多核 SoC 片上跟踪调试技术研究[D]. 北京:中国运载火箭技术研究院, 2020.

Duan H X. Research on On-chip Tracing Debug Technology based on System Bus for Multi-core SoC[D]. Beijing, China: China Academy of Launch Vehicle Technology, 2020.

[8] Arda S, Anish N, Goksoy A A, et al. DS3: a System-level Domain-specific System-on-chip Simulation Framework[J]. IEEE Transactions on Computers, 2020, 69(8):1248–1262.

[9] 魏立强. 一种基于卫星和惯导算法的终端 SDK 解决方案[J]. 长江信息通信, 2021, 34(7):25–27.

Wei L Q. A Terminal SDK Solution based on GNSS and INS Algorithms[J]. Changjiang Information & Communications, 2021, 34(7):25–27.

[10] 刘东明, 王仁平, 李宏意, 等. 微处理器芯片的层次化综合与物理设计[J]. 有线电视技术, 2019, 26(1):89–92.

Liu D M, Wang R P, Li H Y, et al. Hierarchical Synthesis and Physical Design of Microprocessor Chip[J]. CATV Technology, 2019, 26(1):89–92.

[11] 王洁茹. ARM 双核处理器芯片的层次化物理设计方法研究[D]. 西安: 西安电子科技大学, 2022.

Wang J R. Research on Hierarchical Physical Design Method of ARM Dual-core Processor Chip[D]. Xi'an, China: Xidian University, 2022.