

使用 SystemC 设计片上自演化系统

王晓艳, 姚 睿

(南京航空航天大学 自动化学院, 江苏 南京 210016)

摘 要: 提出片上自演化系统的概念和基于 SystemC 的片上自演化系统设计方法, 给出片上自演化系统的总体结构, 使用 SystemC 建模搭建自演化系统实验平台. 以典型低通切比雪夫滤波器为例, 验证了实验平台的有效性. 使用 SystemC 设计自演化系统既可在较高的抽象水平搭建自演化系统模型, 加速验证、性能分析和探索系统结构, 又可方便地进行软硬件协同设计, 并最终达到硬件实现.

关键词: SystemC; 进化硬件; 片上系统; 遗传算法; 自主演化

中图分类号: TP301.6

文献标识码: A

文章编号: 1000-7180(2011)06-0022-04

Self-evolving Design of System on a Chip Using SystemC

WANG Xiao-yan, YAO Rui

(College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

Abstract: The concept of self-evolving SoC and the method of self-evolving design of SoC based SystemC are proposed. , the overall structure is given , and built self-evolving SoC test platform using SystemC modeling. Typical low-pass Chebyshev filter as an example to verify the validity of the experimental platform. Using SystemC design self-evolving SoC, can be on the higher abstraction level built the model of self-evolving system, accelerate validation, run performance analysis and explore system architecture, eventually realize hardware.

Key words: SystemC; evolvable hardware; SoC; genetic algorithm; self-evolving

1 引言

进化硬件(Evolvable Hardware, EHW)是一种像生物一样具有自修复、自适应、自组织特性的特殊硬件,它可以根据使用环境的变化而改变自身结构以适应其生存环境^[1]. 而以 EHW 为主要组成部件的自演化系统是以进化算法特别是遗传算法作为组合优化和全局搜索的主要工具,将可编程器件作为主要的评估手段和实现载体,试图在不依赖先验知识和外力推动(如人工干预)的条件下,通过进化来寻求满足给定要求的电路和系统结构,进而使系统能自动地、实时地调整(重新配置)其内部结构,以适应内部条件(如局部故障)和外部环境(功能要求或物理条件)的变化^[2].

进化硬件与传统硬件的不同在于它具有自组

织、自适应和自修复能力. 目前,国内外大多数进化硬件研究人员(尤其是国内)主要研究进化设计技术,即利用个人电脑(Personal Computer, PC 机)的工作环境进行仿真实验,以验证评估进化硬件的基本原理与改进进化设计方法. 然而,要将进化硬件付诸工程应用必须实现硬件的在线自适应与自修复,其前提是能够实现能够自演化的进化硬件系统. 要实现自主演化的嵌入式数字系统,必须使系统能够脱离 PC 机的工作环境独立工作^[3]. 近年来,SoC 设计与实现方法的研究已经成为微型计算机系统的热点. 因此,若将进化控制和可重构硬件在一个芯片中实现,位流配置和数据通信都可以在片内完成,即可达到最快的速度和最大的灵活性,并进一步构成能够自适应和自修复的单芯片进化系统.

本文提出的片上自演化系统由中央处理器(简

收稿日期: 2010-09-13; 修回日期: 2010-12-01

基金项目: 南京航空航天大学基本科研业务费专项科研项目(NS2010082)

称 CPU)、进化算法硬件加速器、存储器等部分组成.若采用传统的设计方法(由系统规范,系统分割,到硬件与软件分开设计.硬件使用硬件描述语言如 VHDL 或者 Verilog HDL,软件使用 C、C++ 或者汇编语言实现.),不仅存在使用不同语言进行系统描述的不一致性,而且不利用进行软硬件协同设计.而基于 SystemC 语言的设计方法仅使用一种语言就可以完成从系统到寄存器传输级(Register Transfer Level, RTL)、从软件到硬件的全部设计,同时满足了系统复杂度和产品上市时间的要求^[4].

本文提出了基于 SystemC 的片上自演化系统设计方法,给出了自演化系统的总体结构,采用精简指令集(Reduced Instruction Set Computer, RISC)处理器作为进化控制器,将遗传算法作为硬件加速模块,使用 SystemC 建立了自演化系统模型,使用开放源代码的指令集仿真器(Instruction Set Simulator, ISS)建立了自演化系统仿真平台.并以典型低通切比雪夫滤波器为例,验证了实验平台的有效性.

2 SystemC 建模平台

SystemC 是 OSCI(Open SystemC Initiative)组织于 1999 年开始制定和维护的一种基于 C++ 的系统级描述语言,它本质上是在 C++ 的基础上添加的硬件扩展库和仿真核,支持门级、RTL、交易级、系统级等各个抽象层次上硬件的建模和仿真,而且是开放源代码. SystemC 源代码可以在任何标准的 C++ 编译环境下编译(比如 VC++ 6.0, System_Win),可以生成可执行文件;一旦运行可执行文件可生成 VCD、WIF、ISDB 格式的波形文件,可同时查看编译结果和波形文件,而不需要再安装其他的波形查看工具^[5-8].

SystemC 仿真平台提供的建模元素主要有:模块、进程、端口、信号和时钟等.

- 模块:模块是描述设计对象结构的最基本单位,采用 SC_MODULE 来声明模块,描述模块可采用 SC_MODULE 模块类.

- 进程:进程是一个基本执行单位,可用来表达并发行为,而且进程对指定信号和端口敏感,当敏感表中的信号和端口发生变化,就会相应执行此进程. SystemC 的基本进程有三种:方法进程(SC_METHOD)、线程进程(SC_THREAD)、钟控线程进程(SC_CTHREAD).它们分别应用在不同的场合,一起完成对系统对象的描述.

- 端口:模块通过端口与其他模块进行通信.

SystemC 支持输入(input)、输出(output)和输入输出双向端口(inout).

- 信号:信号用于实现进程间通信,信号就是数据的载体.

- 时钟:在 SystemC 中,时钟作为一个 SC_CLOCK 类处理,是仿真过程中系统的计时器.

为了支持系统级设计, SystemC 还拥有许多更高层次的建模特性,如通道、接口和事件.拥有这些建模特性可以对系统的抽象通信机制和同步机制建模,它支持基于接口的设计方法.丰富的数据类型也是 SystemC 的特点之一.它具备丰富的数据类型并支持多种抽象级别和设计领域.定点数据类型可用于数字信号处理(Digital Signal Process, DSP),而固定精度数据类型用于快速模拟,在大量数据计算中可使用任意精度类型. SystemC 有一个基于循环的仿真核,可以进行高速的仿真.其他的属性还包括多重抽象级别,通讯协议,单纠错支持和波形跟踪.

3 自演化系统的 SystemC 建模

3.1 自演化系统总体结构

本文提出的片上自演化系统如图 1 所示.整个实验平台主要包括两大部分:RISC CPU(图 1 中虚线框)和遗传算法硬件加速器(Genetic Algorithm, GA). CPU 模块的功能是实现寄存器之间的存取操作、运算操作、控制 GA 模块. GA 主要完成演化控制.

3.2 SystemC 建模

本文使用 SystemC 建立了自演化系统模型,该模型本质上是一个基于 SystemC 的可执行程序.整个系统的功能在顶层模块(TopModel)中实现.将 CPU 和 GA 作为 SystemC 中的模块在顶层模块中例化,它们之间靠 SystemC 信号进行连接,如图 2 所示.这两个模块并行执行,由时钟统一各进程间的时序.

3.2.1 RISC CPU 模型

CPU 在 SoC 中担负着指令控制、操作控制和数据加工等功能. RISC CPU 由于指令集简单,结构相对固定,逐渐成为片上系统和嵌入式系统的主流处理器.本文建立的 RISC CPU 模型如图 3 所示,主要由取指令、解码与控制、数据管理、指令执行、中断、内存管理等模块组成.

取指令模块 Fetch 完成存储器访问地址的更新,实现程序计数器 PC 的自加,接收来自译码 Decode、执行 Execute 模块的信号以实现 Branch、

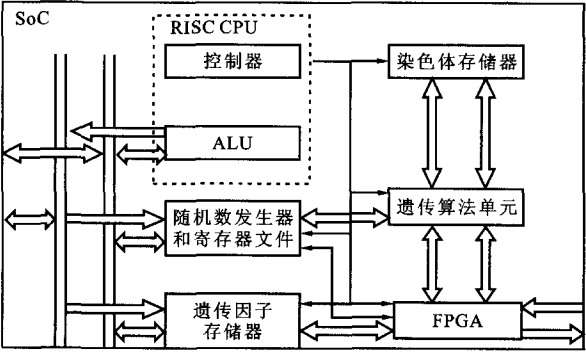


图 1 自演化片上系统结构框图

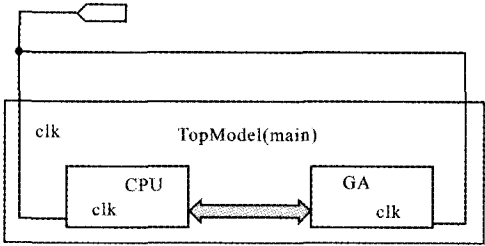


图 2 顶层模块互联图

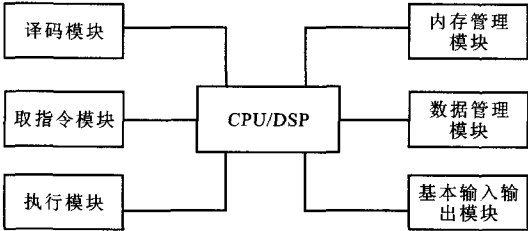


图 3 RISC CPU 结构图

Load/store 指令时取指地址的变化以及相应 PC 的替换。

Decode 模块主要实现指令译码、控制转移指令的目标地址计算、对内部通用/专用寄存器的读写和 Load/store 指令的数据传输等功能。该模块是 RISC 的核心。

Execute 模块根据指令操作码的不同，对操作数进行算术/逻辑/移位运算。数据管理模块做为存储器使用，辅助其他模块工作。

该 CPU 的工作流程如图 4 所示。在基本输入输出模块里，指令存于指令缓存中。当基本输入输出模块的 bios_valid 端口有效时，通过信号的传递，Fetch 模块的 bios_valid 端口有效，开始取指令操作。取到指令后置 instruction_valid 端口为 1。由于端口是由信号连接的，故信号 instruction_valid 的值也为 1。Decode 模块检测到与 instruction_valid 信号连接的 instruction_valid 端口为 1 时，就进行解码操作，解码完成后置端口 decode_valid 为 1。把解码后的操作码和操作数读入 Execute 模块，执行

相应的操作。可以对整数和浮点数进行操作。如果要解码的指令较多时，可以把数据缓存，完成此功能的是数据管理模块。

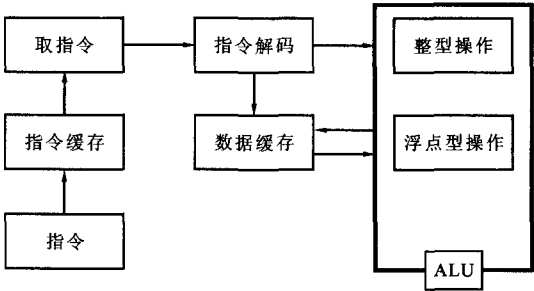


图 4 RISC CPU 的工作流程图

3.2.2 硬件加速器即遗传算法(GA)模块

遗传算法为 EHW 提供了理论和方法学基础。它是由生物界的进化规律演化而来的随机化搜索方法。它模拟自然界物种的繁殖、交叉和基因突变现象，在每次迭代中按某种指标从解种群中选择较优个体，再利用交叉和变异算子对这些个体进行从新组合，产生新一代候选解群，重复此过程，直到满足某种收敛指标为止。要实现自演化功能，片上自演化系统必须要有 GA 模块。

本文使用 SystemC 在行为级建立遗传算法模型，该模型包括遗传算法单元、遗传因子存储器、随机数发生器和寄存器文件、染色体存储器。将种群数、遗传算子分别放在不同的存储器中，用户可以很方便的读取和修改。随机数发生器由函数 rand() 代替。该模块工作流程如图 5 所示。

首先，一个染色体种群在随机数发生器中随机产生，计算种群个体的适应度，一组新的样本种群从前代中最好的样本池产生，准备进行新一轮迭代。迭代之前要进行两个操作：随机交换这些个体的染色体的相关位，即交叉操作，和随机改变染色体位，即突变操作。这一过程重复 N 代，以产生越来越好的个体。整个流程是一个大的循环，如果得到最优解就退出循环并保存适应度值和最优解到染色体存储器中；如果没有得到最优解就继续循环，本程序的循环终止条件是最大代数。

4 实例验证

本文设计的片上自演化系统模型已经仿真实现。因以低通滤波器为基础，经频率变换可以得到各种类型的滤波器，所以下面以二阶低通切比雪夫滤波器的演化为例验证实验平台的有效性。滤波器的演化本质上是优化滤波器的控制参数，进而设计出

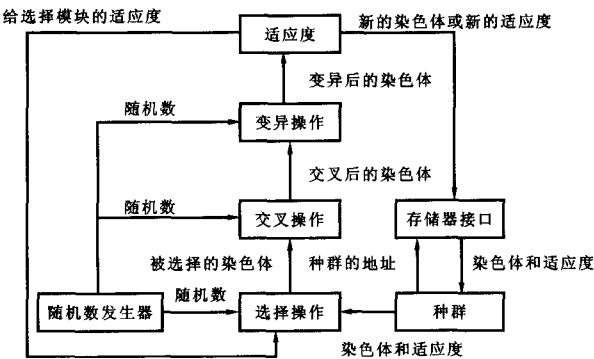


图 5 遗传算法模块的流程图

滤波器.

4.1 二阶低通切比雪夫滤波器模型

二阶切比雪夫滤波器的电路图如图 6 所示.

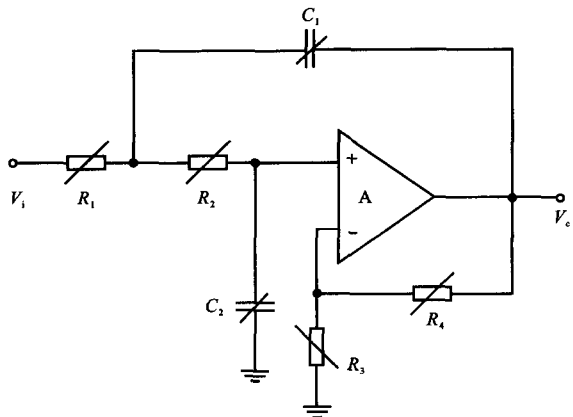


图 6 二阶低通切比雪夫滤波器

由图 6 知,其传递函数 T 为

$$T = \frac{K}{R_1 R_2 C_1 C_2 s^2 + [R_2 C_2 + R_1 C_2 - (1 - K) R_1 C_1] s + 1}$$

(1)

其中 $K = 1 + \frac{R_4}{R_3}$.

4.2 设计步骤

取切比雪夫滤波器的截止频率为 f_c 、增益为 K 、纹波为 δ . 具体步骤如下:

(1) 确定待优化的变量:滤波器的增益 K 由 R_3 、 R_4 决定. K 与滤波器的通带没有关系,所以把 R_3 、 R_4 设为定值. 设 R_3 为 1 k Ω 、 R_4 为 5 k Ω ,所以 K 的值为 6. 算法中只要优化 R_1 、 R_2 、 C_1 、 C_2 这 4 个参数即可.

(2) 建立优化模型:从 $[0, \omega_p](\omega_p > \omega_s, \omega_s$ 为阻带边界频率) 中均匀取 10 个频率点,把 4 个元件参数带入式(1),计算出每个频率点处的传递函数的幅值;利用仿真平台中的 GA 不断修正元件参数,使电路的传递函数满足设计要求.

(3) 编码方法:采用实数编码方案. 将编码取为

$X_i = [R_1, R_2, C_1, C_2]$, 取值范围为: $R_1 = 0 \sim 205.48 \Omega$, $R_2 = 0 \sim 102\,300 \Omega$, $C_1 = 0 \sim 127\,000$ nF, $C_2 = 0 \sim 150.627$ nF, 个体样本是随机产生的.

(4) 适应度函数:适应度函数的好坏对进化算法的成功起到关键的作用,而目前为止对适应度函数的取法也没有统一的规定,但一般适应度函数总是使其适应度值越大越好. 本例中,构造了一个简单的适应度函数:

$$f(x) = \frac{1}{\sum_{f \leq 100} (K_1 - K)^2 + \sum_{f \geq 100} (K_2)^2}$$

(2)

式中, K_1 是通带内滤波器的幅值, K_2 为阻带内滤波器的幅值, K 是滤波器的幅值,本例中 K 的值为 6, 适应度函数越大表明逼近效果越好,滤波器的滤波效果越好.

4.3 实验结果

取种群数为 50, 运行代数数为 500, 交叉率为 0.7, 变异率为 0.1. 运行 30 次, 仿真得到了二阶低通切比雪夫滤波器的优化参数. 如表 1 所示.

表 1 优化结果

最大适应度值	最优解			
	R_1/Ω	R_2/Ω	C_1/nF	C_2/nF
0.374 531 835 2	1.027	63 221.4	2 540.0	143.698

根据表 1 给出的电路优化参数, 绘出滤波器幅频特性图(图 7).

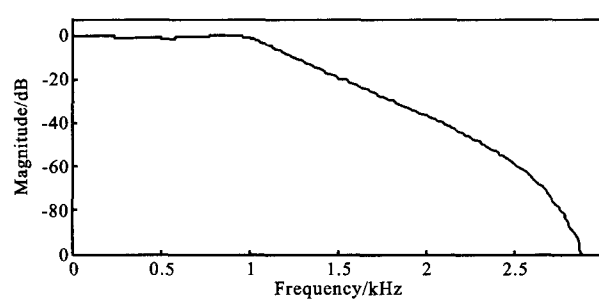


图 7 滤波器幅频特性图

从图 7 可以看出, 滤波器通带、阻带特性良好, 但是过渡带稍宽, 滤波效果不是很好.

5 结束语

本文提出了片上自演化系统的概念和基于 SystemC 的片上自演化系统设计方法. 使用 SystemC 建立了片上自演化系统模型, 包括 CPU 模块和 GA 模块, 由 CPU 进行计算和控制, GA 完成演化控制. 并以典型低通切比雪夫滤波器为例, 验证了实验平台的有效性. (下转第 30 页)

宏块预测与边界滤波强度计算算法的 VLSI 架构。通过利用算法之间的共性与依赖关系,该架构最大程度的复用硬件设计和减少存储空间,并达到高效的解码性能。该结构不仅能够支持 H. 264 1080p@HiP 和 AVS 1080p@JiZhun Profile 实时处理的需求,还能应用于当前的可重构多媒体系统,提高系统的解码性能。综合和仿真后的结果表明,该结构在面积和性能上均优于当前类似文献的结果,远远满足 H. 264 和 AVS 高清实时处理的需求。

参考文献:

- [1] Geng T S, LIU Leibo, Yin Shouyi, et al. Parallel implementation of computing intensive decoding algorithms of H. 264 on reconfigurable SoC[C]// Proc of 2010 IEEE International Symposium Circuits and Systems (ISCAS). Paris: IEEE, 2010:1153-1156.
- [2] Zhu M, Liu Leibo, Yin Shouyi, et al. A reconfigurable multi-processor SoC for media applications[C]// Proc of 2010 IEEE International Symposium, Circuits and Systems (ISCAS). Paris: IEEE, 2010:2011-2014.
- [3] 黄玄, 陈杰, 李霞, 等. AVS 高清视频变长码解码器算法与电路实现[J]. 微电子学与计算机, 2009, 26(1):76-80.
- [4] Yin H B, Zhang D P, Wang Xiumin, et al. An efficient MV prediction VLSI architecture for H. 264 video decoder[C]// 2008 International Conference on Audio, Language and Image Processing, Shanghai: ICALIP, 2008: 423-428.
- [5] Zheng J H, Wu D, Deng L, et al. A motion vector predictor architecture for AVS and MPEG-2 HDTV decoder[J]. Journal of Computer Science and technology, 2006, 21(3): 424-431.
- [6] 谢朝辉, 冯燕, 李谦, 等. H. 264 和 AVS 多模视频解码器中运动矢量预测的硬件[J]. 微电子与计算机, 2006, 23(11):54-57.
- [7] Dang P. High performance architecture of an application specific processor for the H. 264 deblocking filter[J]. IEEE Trans on VLSI Systems, 2008, 16 (10): 1321-1334.

作者简介:

陶昱良 男, (1985-), 硕士研究生. 研究方向为视频编解码处理的 ASIC 和 SoC 设计, 高性能可重构运算阵列架构。

何卫锋 男, (1978-), 博士, 副教授. 研究方向为视频编解码系统的 VLSI 设计, SoC 设计方法学与系统集成。

王 琴 女, (1975-), 博士, 副教授, 研究方向为音视频处理应用开发 SoC, 高性能低功耗体系架构。

(上接第 25 页)

该自演化系统模型是在较高的抽象水平建立的, 可用以加速验证、性能分析和系统结构的探索。目前, 整个系统在行为级已经仿真实现, 后期的工作可以综合细化到 RTL, 使整个系统能在现场可编程门阵列(Field Programmable Gate Array, FPGA)上实现。此系统还可以增加其他的模块, 比如: 通用串口, 通用 I/O, 总线等模块以完成更复杂的功能。

另外, 由于目前自演化系统中演化控制采用的是简单的标准遗传算法, 滤波效果不是很好。后期可进一步调整遗传算法控制参数, 并改进遗传算法, 以提高算法效率和搜索结果。

参考文献:

- [1] 姚睿, 王友仁, 于盛林, 等. 具有在线修复能力的强容错三模冗余系统设计及实验研究[J]. 电子学报, 2010, 38 (1):177-183.
- [2] Yao Rui, Yu Shenglin, Wang Youren, et al. Online evolution of the algorithm-level digital hardware system using evolvable core[J]. Journal of Chinese Computer Systems, 2009, 30(1):188-192.
- [3] 陈绍贺, 赵明, 王京. 基于 SystemC 的片上系统设计[J]. 微电子学与计算机, 2005, 22(4):51-52.
- [4] 李烨挺, 梁利平. 一种基于 SystemC 的片上网络建模与仿真方法[J]. 微电子学与计算机, 2010, 27(3):78-82.
- [5] Sekanina L. Virtual reconfigurable circuits for real-world applications of evolvable hardware[C]// Proc of the 5th Int Conf on Evolvable Systems: From Biology to Hardware (ICES'03), Trondheim, Norway: Springer, 2003:186-197.
- [6] Chen Xi, Xu Ningyi. SystemC SoC design[M]. Beijing: Science Press, 2004.
- [7] SystemC. [EB/OL]. [2010-09-13]. <http://www.systemc.org>.
- [8] SystemC version2.0 User Guide. Synopsys, Inc. [EB/OL]. (2002)[2010-09-13]. <http://www.systemc.org>.

作者简介:

王晓艳 女, (1983-), 硕士研究生. 研究方向为计算机测控、嵌入式系统设计及应用。

姚 睿 女, (1974-), 博士, 副教授. 研究方向为进化硬件理论与技术、嵌入式测控系统自修复。