2016年2月 Journal of Terahertz Science and Electronic Information Technology

文章编号: 2095-4980(2016)01-0136-07

一种 SRAM 型 FPGA 互连资源的位流码配置方法

李智华 a,b, 黄 娟 a, 李 威 b, 杨立群 a,b, 黄俊英 a,b, 杨海钢 a^*

(中国科学院 a.电子学研究所; b.中国科学院大学, 北京 100190)

摘 要:针对静态随机存取存储器(SRAM)型现场可编程门阵列(FPGA)位流码配置问题,提出 一种自动配置互连资源的方法。该方法从描述 FPGA 结构的行为级 Verilog 文件中,采用基于端口 映射的记忆 FPGA 配置模型搜索(MCMS)算法自动提取互连资源的配置位模型,然后结合布线结果 生成布线路径上互连资源的位流码。实验结果表明,对于包含 30 Mb 配置位的 3 000 万门 SRAM 型同质 FPGA,采用人工方法提取互连资源配置位模型需要 6 天时间,而采用端口映射 MCMS 算 法仅需要 29 分钟,效率提高了 298 倍;对于同等规模的异质 FPGA,采用人工方法需要 7 天时 间,而采用端口映射 MCMS 算法仅需 26 分钟,效率提高了 394 倍。该算法作为一种通用的互连 资源配置位模型提取方法,可以应用于不同的 FPGA 芯片。在缩短位流码配置时间的同时,提高 位流码配置的准确性。

关键词:现场可编程门阵列;位流码配置;互连资源配置位模型;MCMS算法
中图分类号:TN47
文献标识码:A
doi: 10.11805/TKYDA201601.0136

An automatic approach for bitstream configuration of routing resource in SRAM FPGA

LI Zhihua^{a,b}, HUANG Juan^a, LI Wei^a, YANG Liqun^{a,b}, HUANG Junying^{a,b}, YANG Haigang^{a*} (a.Institute of Electronics; b.University of Chinese Academy of Sciences, Chinese Academy of Sciences, Beijing 100190, China)

Abstract: A Memorized Field Programmable Gate Array(FPGA) Configuration Model Search(MCMS) algorithm is presented to automatically generate configuration bitstream of routing resources to solve the bitstream configuration problem commonly existing in Static Random Access Memory(SRAM) FPGA. The main objective is to reduce the time for generating configuration bitstream of routing resources and improve the correctness of configuration process. This is achieved by allowing the MCMS algorithm to extract the model of configuration bitstream from the behavioral Verilog HDL(Hardware Design Language) netlist that describes the FPGA architecture, utilizing the model to automatically generate the bitstream of routing resource on the routing path based on the routing result. Our experimental results using a homogeneous SRAM FPGA with 30 million gates that contains 30 million configuration bits have shown a 298 times improvement in time efficiency, from 6 days when manually extracting the configuration bitstream model to 29 minutes when using MCMS to complete the extraction. For a heterogeneous FPGA of the same scale, it takes one week for the manual method to extract the configuration bitstream model, while the MCMS only needs 26 minutes, which improves the time efficiency by 394 times. By using the automatic method, not only the time for generating configuration bitstream can be shortened, but also the correctness of configuration bitstream generated can be ensured. As a general configuration bitstream model extracting procedure, the proposed approach is applicable for homogeneous and heterogeneous FPGAs.

Key words: Field Programmable Gate Array(FPGA); bitstream configuration; model of configuration bitstream; Memorized FPGA Configuration Model Search(MCMS)

基于 SRAM 的 FPGA 具有可重复编程、开发周期短、开发成本低等特点,因此在过去 30 年得到了飞速的 发展和广泛的应用^[1]。为了辅助用户进行电路设计,出现了 FPGA 计算机辅助设计(Computer Aided Design, CAD)工具^[2]。FPGA CAD 流程包括综合、工艺映射、装箱、布局、布线和位流码配置^[3]。位流码配置阶段作为 CAD 流程中的关键环节^[4],主要实现对互连资源、基本逻辑单元的逻辑功能以及异质结构 FPGA 中 IP 核 (Intellectual Property Core)功能的可编程控制;其中实现互连资源的配置位数目占总配置位的 70%左右^[5],对其 配置主要依据用户电路的布线结果来实现,而对基本逻辑单元的逻辑功能以及异质结构 FPGA 中 IP 核功能的配 置则是由综合后的电路逻辑功能决定。本文重点研究针对互连资源的位流码配置方法。

随着芯片规模的增大, 位流码的数目也变得更多, 同时, 位流码配置变得更加复杂, 但在学术界中很少有关于高效位流码配置方法的研究。学术界知名的 FPGA CAD 开源工具 VPR^[6]以及支持异质 FPGA 结构的 VTR^[7] 作为 FPGA 架构评估工具, 由于没有形成正式芯片, 并不支持位流码配置。由于 SRAM 配置与 FPGA 结构密切相关, 而商业 FPGA 结构都存在知识产权保护, 因此在工业界,关于位流码配置方法的公开资料也较少。目前 主要的 FPGA 生产商 Altera 公司和 Xilinx 公司, 有关其位流码配置方面的算法都未进行公开。

本文提出了一种高效的自动配置互连资源的方法。该方法首先将描述 FPGA 结构的行为级 Verilog 硬件描述 语言(Hardware Description Language, HDL)文件作为输入,采用端口映射 MCMS 算法提取互连资源配置位模型,再结合各个信号的布线结果,自动生成布线路径上互连资源的位流码。该方法作为通用的互连资源位流码 配置方法,能够应用于多种结构的 FPGA 芯片,极大缩短了位流码配置时间。

1 SRAM 型 FPGA 位流码的配置过程

1.1 SRAM 型 FPGA 结构

SRAM 型同质结构 FPGA 的硬件结构主要有 3 部分:可编程逻辑功能块(Configurable Logic Block, CLB),用于实现用户电路的逻辑功能;互连资源,用于提供逻辑块间的连接关系;输入输出块(IO),用于实现芯片内部信号与外部信号之间的通信^[8]。相比于同质结构 FPGA,异质结构 FPGA 内嵌了多种宏单元模块,如存储器和 DSP 等。同质结构和异质结构 FPGA 的组成如图 1(a)、图 1(b)所示。



Fig.1 FPGA structure 图 1 FPGA 结构

从 FPGA 硬件结构可以看出,同质结构 FPGA 的配置位主要实现互连资源的连通功能和基本逻辑单元的逻辑功能,异质结构 FPGA 的配置位还需要实现宏单元模块功能的可编程控制。其中,互连资源包括连线连接盒 (Connection Box, CB)和连线开关盒(Switch Box, SB),如图 2 所示。CB 将逻辑块的输入和输出端口连接到互连通道上,而 SB 则将水平和垂直方向上的互连通道连接起来。在实际的 FPGA 芯片中,CB 和 SB 均由不同输入的多路选择器(Mutiplexor,Mux)组成。在用 FPGA 实现应用电路时,通过配置相应的位流码使得 CB、SB 中的可编程开关导通,为信号提供通路^[9]。

138

位流码的配置分为 2 部分:一是根据 CAD 工具综合后的结果,配置相应的位流 码实现 CLB 的逻辑功能以及异质结构 FPGA 宏单元的功能;二是根据电路布线 后的信号路径,配置互连资源的位流码, 连通所需的路径。由于互连资源的配置位 占总配置位数目的 70%以上,下面重点介 绍互连资源的位流码配置方法。

多路选择器是互连资源的最小可配置 单元,配置互连资源即配置相应的 Mux。

配置互连资源的位流码需要以下 2 步:1)提取互连资源配置位模型。配置位 模型包括2部分,第1部分是Mux的配置 位在顶层 SRAM 配置位中的对应位。由于 SRAM型 FPGA的配置位以矩阵形式排 列,每个配置位以行号r和列号c进行标 识,记录为[r,c],故Mux的配置位的对应 位也表示为[r,c];第2部分是Mux的配置 位控制输入端的方式。对于n输入Mux, 其配置位有m个,分别为 s_0,s_1, \dots, s_{m-1} ,通



常情况下, $m = \lceil \log_2^n \rceil$, 则此 Mux 的配置位控制输入端的方式表示为 $\langle s_{m-1}, \dots, s_1, s_0 \rangle \langle (d_0, d_1, \dots, d_{n-1}) \rangle$ 。当选通 d_i 时, 配置位 s_{m-1}, \dots, s_1, s_0 为数值等于 *i* 的二进制数。2) 根据布线路径确定信号连接的互连资源中输入端和输出端的位置, 配置相应的位流码连通其中的 Mux。

由图 3 所示,下面以 2 个相同 2 选 1 的 Mux 组成的互连资源为 例,说明互连资源的位流码配置方法。布线路径如图中箭头所示, 信号从 mux_1 的 d_0 端输入,经过 mux_1 输出后到达 mux_2 的 d_1 输 入端,最后通过 mux_2 输出。对其的配置过程为,首先提取 Mux 的配置位模型,假设 mux_1 和 mux_2 的配置位对应的顶层 SRAM 配置位分别为 $[r_1,c_1]$ 和 $[r_2,c_2]$,mux_1 和 mux_2 的配置位控制输入端 的方式均为<s> (d_0,d_1) ;然后,根据布线结果,mux_1 需将 d_0 连 通,所以将 $[r_1,c_1]$ 的顶层 SRAM 配置位设置为 0;mux_2 需将 d_1 连 通,所以将 $[r_2,c_2]$ 的顶层 SRAM 配置位设置为 1,从而完成互连资 源位流码的配置。



Fig.3 Bitstream configuration of routing resources 图 3 互连资源位流码的配置示意图

从上述描述可知,在互连资源位流码的配置过程中,提取互连资源的配置位模型是难点,也是研究的重点。配置位模型提取方法可分为人工提取和自动化提取 2 种。人工提取是指设计人员根据原理图中 Mux 的工作 原理,找到 Mux 的配置位控制输入端的方式及其对应的顶层 SRAM 配置位的行号和列号。这种方法不仅效率 低而且容易出错。自动化提取配置位模型可以用 2 种方法实现。第 1 种方法是运用计算机图像识别技术模拟人 工观察来识别原理图中的底层配置模块。这种方式算法复杂度高,且图像识别的准确度也是一个很大的挑战。 第 2 种方法是将 FPGA 原理图转换成形式化的文字表述,例如 Verilog HDL 行为描述语言。其结构化的特点非 常适合于描述 FPGA 原理图的层次化设计,所以从 Verilog HDL 描述中提取互连资源配置位模型可以降低算法 复杂度,准确识别配置位模型。

本文提出一种高效的自动配置互连资源的方法。该方法将 FPGA 芯片原理图转换为行为级 Verilog HDL 文件^[10],对该文件进行词法和语法分析生成抽象语法树(Abstract Syntax Tree, AST),然后从 AST 中提取针对 FPGA 芯片层次化描述的 Verilog 模块树,在该模块树中,根据信号流向关系,利用端口映射的 MCMS 算法提取互连资源配置位模型,再结合布线结果生成布线路径上互连资源的位流码。

第1期

2 基于 MCMS 算法的互连资源配置方法

图 4 呈现了从描述 FPGA 结构的行为级 Verilog 文件自动完成互连资源配置的方法,首 先对该 Verilog 文件进行词法和语法分析生成 AST,从 AST 中提取 Verilog 模块树,然后根 据信号流向关系,提取互连资源配置位模型。 其中根据信号流向关系,提取互连资源配置位 模型是由 MCMS 算法完成。最后,以提取的配 置位模型为基础,根据 CAD 工具的布线结果, 生成相应的配置位流码。



Fig.4 Flow from FPGA Verilog HDL to bitstream configuration of routing resources 图 4 从 FPGA Verilog 描述生成互连资源配置位流码的流程

2.1 生成 Verilog 模块树

首先,按照 Verilog HDL 描述的巴科斯范式(Backus Naur Form, BNF),使用词法解析工具 Lex^[11]、语法语 义分析工具 Yacc^[11]对描述 FPGA 结构的行为级 Verilog HDL 文件进行处理,并建立抽象语法树^[12]AST。

其次,从 AST 中最顶层的模块开始构建 Verilog 模块树,顶层模块作为树的父节点,在遇到实例化子模块

时便将子模块作为父节点的子节点,然后将子节 点作为父节点,继续添加其子节点,重复上述过 程,直到子节点上的模块为 FPGA 底层门级单 元。在上述提取过程中,需要预设一个堆栈来记 录各个模块的处理情况。

从 AST 中提取的 Verilog 模块树结构如图 5 所示,模块树的根节点对应了 FPGA 芯片原理图 中的顶层模块,其第一级子节点包含了 FPGA 中 不同种类的逻辑块如 CLB、IO、其他异质模块以 及 SRAM 配置模块等,模块树的叶子节点对应了 FPGA 芯片的底层门级单元。



2.2 MCMS 算法流程

Verilog 模块树保存了对 FPGA 芯片原理图进行层次化描述的所有信息。利用模块树中顶层模块和底层模块 之间的信号连接关系,可以找到顶层配置位和底层配置模块端口的映射关系。为此,本文提出了一种从 Verilog 模块树中提取互连资源配置位模型的 MCMS 算法。该算法主要包括 3 个步骤:

1) 使用正则表达式匹配模块名,识别模块树中的底层配置模块。算法从 FPGA 模块树的顶层模块和配置端口开始搜索,如果模块名符合底层配置模块的特征关键字,则直接返回,得到互连资源配置位模型,否则转步骤 2);

2) 搜索和顶层配置端口有信号连接关系的子模块。对于当前模块的所有子模块,如果不存在任何引脚和顶层配置端口连接,则说明配置位引脚悬空,该配置位没有找到对应的互连资源配置位模型;否则转步骤3);

3) 递归搜索子模块的互连资源配置位模型。如果步骤 2)中找到的子模块不包含任何嵌套模块,且该子模块名符合底层配置模块的特征关键字,则直接返回,得到互连资源配置位模型;否则,将子模块作为顶层模块,和顶层配置位端口连接的子模块引脚作为待搜索的配置端口,转步骤 1)递归搜索互连资源配置位模型,并返回搜索结果。

在进行互连资源配置位模型搜索过程中,为了避免同一类型模块的重复搜索,MCMS 算法使用了 Memoization 优化技术^[13-14],使用查找表来存储底层模块中所有配置端口的互连资源配置位模型。在自顶向下 的模块树搜索过程中,如果发现子模块的配置端口对应的互连资源配置位模型包含在查找表中,则直接获取该 配置位模型,否则进入子模块中搜索。实验证明,这种优化技术可使 MCMS 算法的运行效率提高 30 倍左右。

2.3 基于端口映射的 MCMS 算法实现配置位模型提取

MCMS 算法的性能瓶颈在于随着 FPGA 芯片规模的增大,算法的运行时间大部分都消耗在搜索和顶层配置端口有信号连接关系的子模块上。以 CLB 阵列规模为 100×100 的同质 FPGA 芯片为例,一个 CLB 包含 60 个端口,平均每个端口名大约包含 5 个字符。为了找到和 SRAM 上某一配置端口相连的 CLB 端口, MCMS 算法需

遍历所有 CLB 端口来判断信号的连接关系,完成这 步操作需要搜索的字符数总计为 300 万个 (100×100×60×5 = 3 000 000,约 3 MB字符)。在 CPU 频率为 1.8 GHz 的 PC 上对 linux grep 命令进行实 验,发现在 3 MB 的文本文件中搜索某一特定字符串 的平均时间为 0.005 秒。上述规模的 FPGA 芯片中约 有 1000 万个 SRAM 配置位,搜索和 SRAM 配置位连 接的 CLB 端口需花费的总时间为 13.89 小时。如果 再加上底层模块的配置位搜索时间,算法的运行时间 会更长。

为了避免繁琐的字符串搜索工作,本文通过定义 SRAM 和逻辑块之间的配置端口映射规则来解决这个 问题。能够采用这种方法的原因是 FPGA 芯片的逻辑 块类型是一定的,并且 FPGA 芯片原理图设计人员在 设计 SPAM 配置端口和顶层逻辑也配置端口的连接时



Fig.6 Mapping between SRAM and logic block configuration port 图 6 SRAM 配置端口和逻辑块配置端口的映射关系示意图

设计 SRAM 配置端口和顶层逻辑块配置端口的连接时,通常会遵循一定的映射规则。

图 6 示意了一个同质结构 FPGA, 假设单个 CLB 中的配置位流码为 n 行、m 列,则(x,y)处 CLB 上配置端口 sram1[1]对应的顶层配置位为[(y-1)n+1, (x-1)m+1],其右边的 CLB 位置为(x+1,y),其配置端口 sram1[1]对应的顶层配置位为[(y-1)n+1, xm+1];其上方的 CLB 位置为(x, y+1),其配置端口 sram1[1]对应的顶层配置位为[yn+1, (x-1)m+1]。基于上述映射规则,通过简单的迭代计算,即可得到 FPGA 中任意位置的 CLB 配置端口对应的顶层配置位。对于 IO,DSP,Memory 等其他类型的逻辑块,也能定义类似的映射规则。

2.4 互连资源配置位流码的生成

基于 2.3 中所得的模型,根据 CAD 布线路径确定信号连接的互连资源中输入端和输出端的位置,通过本文 1.2 节描述的方法,配置相应的位流码连通底层配置模块。

3 实验结果

利用所提出的互连资源配置方法,本文进行了一系列实验,将提出的 MCMS 算法和传统的人工提取方法进行比较。基于两种方法分别对不同规模的同质和异质结构 FPGA 进行互连资源配置位流码生成。MCMS 算法运行环境为 Intel 酷睿 2 E7500 双核处理器,处理器频率为 2.93 GHz,机器内存为 3 GB。

3.1 同质 FPGA 上的实验结果

实验中所用的不同规模的同质 FPGA 配置信息统计见表 1,表中包含了 3 万门到 3 000 万门的 FPGA 配置位数目和底层配置模块数目。由该表可以看出,配置位数目随着 FPGA 芯片规模呈线性增长关系;底层配置模块数目也和 FPGA 芯片规模呈正比。

Table1 Statistics of homogeneous FPGA configuration with different scales				
FPGA scale (ten thousand gate-units)	number of configuration bits	number of configuration modules		
3	43 575	24 312		
30	408 515	227 925		
300	3 528 084	1 909 434		
3 000	31 164 742	16 866 667		

表 1 不同规模的同质 FPGA 配置信息统计

对于不同规模的同质 FPGA, 分别采用 MCMS 算法、基于端口映射的 MCMS 算法和人工方法提取互连资 源配置位模型所花费的时间见表 2。

农ZMCMD并每个至于缅甸或加州MCMD并每个的人工力每时间比较						
Table2 The comparison of execution time between MCMS, port-mapping MCMS and artificial method						
FPGA scale (ten	artificial methods	MCMS algorithm	port mapping MCMS	artificial / MCMS	artificial / port mapping	
thousand gate-units)	(days × person)	(minutes)	algorithm (minutes)		MCMS	
3	3.0	2.9	0.27	1 450	16 000	
30	3.5	27.4	0.30	184	16 800	
300	4.5	241.0	3.20	27	2025	
3 000	6.0	2 125.2	28.97	4	298	

表 2 MCMS 算法、基于端口映射的 MCMS 算法和人工方法时间比较

由表 2 可以看出,采用人工方法提取互连资源配置模型,即便是规模为 3 万门的 FPGA 芯片,提取时间也 需要 3 天(按照单人工作计算)。而采用 MCMS 算法的提取时间为 2.9 分钟,为人工方法的 1/1 450;采用基于端 口映射的 MCMS 算法运行时间仅为 0.27 分钟,为人工方法的 1/16 000,较 MCMS 算法运行效率提高了 11 倍。 当 FPGA 芯片规模从 3 万门增长到 3 000 万门时,采用 MCMS 算法提取互连资源配置位模型的运行效率是人工 方法的 4~1 450 倍,采用基于端口映射的 MCMS 算法的运行效率是人工方法的 298~16 000 倍。基于端口映射的 MCMS 算法较 MCMS 算法更快,其原因是基于端口映射的 MCMS 算法仅需要提取一个位置上的逻辑块配置端 口的互连资源配置位模型,其他位置上的逻辑块和 SRAM 配置端口的映射可以通过简单的线性函数计算得到。

3.2 异质 FPGA 上的实验结果

本节实验采用的异质 FPGA 包含了 Memory 和 DSP 宏单元模块,不同规模的异质 FPGA 配置信息统计见表 3,表中包含了 3 万门到 3 000 万门的 FPGA 配置位数目和底层配置模块数目。和表 1 中同等规模的同质 FPGA 芯片相比,FPGA 配置位数目和底层配置模块数目略有减少,原因是 DSP 和 Memory 包含的配置位数目较同等 规模的 CLB 少。

Table3 Statistics of heterogeneous FPGA configuration with different scales					
FPGA scale (ten thousand gate-units)	number of configuration bits	number of configuration modules			
30	384 278	212 414			
300	3 184 017	1 746 181			
3 000	26 645 194	14 468 356			

表 3 不同规模的异质 FPGA 配置信息统计

对于不同规模的异质 FPGA, 分别采用 MCMS 算法、基于端口映射的 MCMS 算法和人工方法提取互连资 源配置位模型所花费的时间见表 4。

表 4 MCMS 算法、基于端口映射的 MCMS 算法和人工方法时间比较

Table4 Comp	arison of execution	on time between	n MCMS,	port-mapping	MCMS and artificial method
-------------	---------------------	-----------------	---------	--------------	----------------------------

FPGA scale (ten	artificial methods MCMS algorithm		port mapping MCMS algorithm	artificial / MCMS	artificial / port mapping
thousand gate-units)	(days × person)	(minutes)	(minutes)	artificiar / IviCiviS	MCMS
30	4	26.0	0.28	222	20 571
300	5	217.7	2.98	33	2 416
3 000	7	1 887.1	25.58	5	394

由表 4 数据可以看出,与表 2 中同等规模的同质 FPGA 相比,人工方法提取互连资源配置位模型的时间略 有增加,原因在于异质 FPGA 中 DSP 和 Memory 模块的引入使得人工提取方法变得更加复杂。而 MCMS 算法 和基于端口映射的 MCMS 算法的运行时间较同质 FPGA 有所减少,这是由于 FPGA 配置位数目和底层配置模块 数目减少的缘故。当 FPGA 芯片规模从 30 万门增长到 3 000 万门时,采用 MCMS 算法提取互连资源配置位模 型的运行效率是人工方法的 5~222 倍,采用基于端口映射的 MCMS 算法的运行效率是人工方法的 394~20 571 倍。基于端口映射的 MCMS 算法效率是 MCMS 算法的 73~92 倍。

位流码配置时间包括互连资源配置位提取时间和根据布线结果生成位流码的时间。而即使当应用电路互连资源的利用率达到 90%时,根据布线结果生成位流码所需的时间也不到 1 s,所以位流码配置时间主要由互连资源配置位提取时间决定。本文提出的基于端口映射的 MCMS 算法可以大大缩短互连资源位流码配置时间。

4 结论

本文提出了一种高效的 FPGA 互连资源位流码配置方法。通过对描述 FPGA 结构的行为级 Verilog 文件的解析,生成 Verilog 模块树。再利用基于端口映射的 MCMS 算法提取位流码模型,最后根据布线结果生成互连资源的配置位流码。试验结果表明,相比于人工方法,利用基于端口映射的 MCMS 算法生成配置位流码,效率最多能够提高两万多倍。所提出的方法可以应用于不同结构、各种规模的 FPGA 芯片。在缩短配置位流码生成时间的同时,提高了位流码配置的准确性。

参考文献:

- [1] 谭斌,刘平净,李锋. 基于 FPGA 的指纹预处理技术[J]. 太赫兹科学与电子信息学报, 2010,8(2):190-195. (TAN Bin, LIU Pingjing,LI Feng. Fingerprint image preprocessing based on FPGA[J]. Journal of Terahertz Science and Electronic Information Technology, 2010,8(2):190-195)
- [2] BETZ V, ROSE J, MARQUARDT A. Architecture and CAD for deep-submicron FPGAs[M]. Kluwer Academic Publishers, 1999.
- [3] 古海云,李丽,许居衍,等. 一种 Virtex 系列 FPGA 配置数据无损压缩算法[J]. 计算机研究与发展, 2006,43(5):940-945. (GU Haiyun,LI Li,XU Juyan, et al. Lossless configuration bitstream compression for Virtex FPGAs[J]. Journal of Computer Research and Development, 2006,43(5):940-945.)
- [4] SRINIVASAN S,GAYASEN A,VIJAYKRISHNAN N,et al. Improving soft-error tolerance of FPGA configuration bits[C]// 2004 IEEE/ACM International Conference on Computer Aided Design. San Jose,CA,USA:IEEE, 2004:107-110.
- [5] 孙劼,童家榕. 层次式布线资源 FPGA 连线开关的设计[J]. 微电子学, 2005,35(4):404-408. (SUN Jie,TONG Jiarong. Design of connection switch for hierarchical field programmable gate arrays[J]. Microelectronics, 2005,35(4):404-408.)
- [6] BETZ V,ROSE J. VPR: A new packing, placement and routing tool for FPGA research[C]// Field-programmable logic and applications. Heidelberg:Springer, 1997:213-222.
- [7] ROSE J,LUU J,YU C W,et al. The VTR project: architecture and CAD for FPGAs from verilog to routing[C]// Proceedings of the ACM/SIGDA international symposium on Field Programmable Gate Arrays. Monterey, California: ACM, 2012:77-86.
- [8] ROSE J,El Gamal A,Sangiovanni-Vincentelli A. Architecture of field-programmable gate arrays[J]. Proceedings of the IEEE, 1993,81(7):1013-1029.
- [9] GRAHAM P,CAFFREY M,ZIMMERMAN J,et al. Consequences and categories of SRAM FPGA configuration SEUs[C]// Proceedings of the 6th Military and Aerospace Applications of Programmable Devices and Technologies International Conference. Washington DC:National Aeronautics and Space Administration, 2003.
- [10] DIMITRIJEVIĆ M,JOVANOVIĆ B,ANĐELKOVIĆ B,et al. Experiences in using CADENCE-the industry standard for integrated circuits design[C]// Proc. XLVII ETRAN Conference. Herceg Novi,Montenegro:[s.n.], 2003.
- [11] LEVINE J R, MASON T, BROWN D. Lex & Yacc[M]. Cambridge, Massachusetts:O'Reilly Media, Inc., 1992.
- [12] LAM M S,SETHI R,ULLMAN J D. Compilers: Principles, Techniques, & Tools[M]. Pearson/Addison Wesley, 2007.
- [13] NORVIG P. Techniques for automatic memoization with applications to context-free parsing[J]. Computational Linguistics, 1991,17(1): 91-98.
- [14] ACAR U A, BLELLOCH G E, HARPER R. Selective Memoization[M]. Washington DC: ACM, 2003.

作者简介:



李智华(1986-),男,四川省南充市人, 在读博士研究生,主要研究方向为大规模集 成电路设计自动化技术.email:76194688@ qq.com.

黄俊英(1986-),女,安徽省淮北市人,在读博士研究 生,主要研究方向为大规模集成电路设计自动化技术.

杨海钢(1960-),男,上海市人,博士,研究员,博士生导师,主要研究方向为高速可编程逻辑芯片设计技术;数模混合信号 SOC 设计技术.

黄 娟(1983-), 女, 湖北省襄阳市人, 博士, 助理研究员, 主要研究方向为大规模集 成电路设计自动化技术.

李 威(1983-),女,黑龙江省大庆市 人,博士,助理研究员,主要研究方向为可编 程逻辑芯片设计.

杨立群(1989-),女,黑龙江省五常市 人,在读博士研究生,主要研究方向为可编程 逻辑芯片设计.