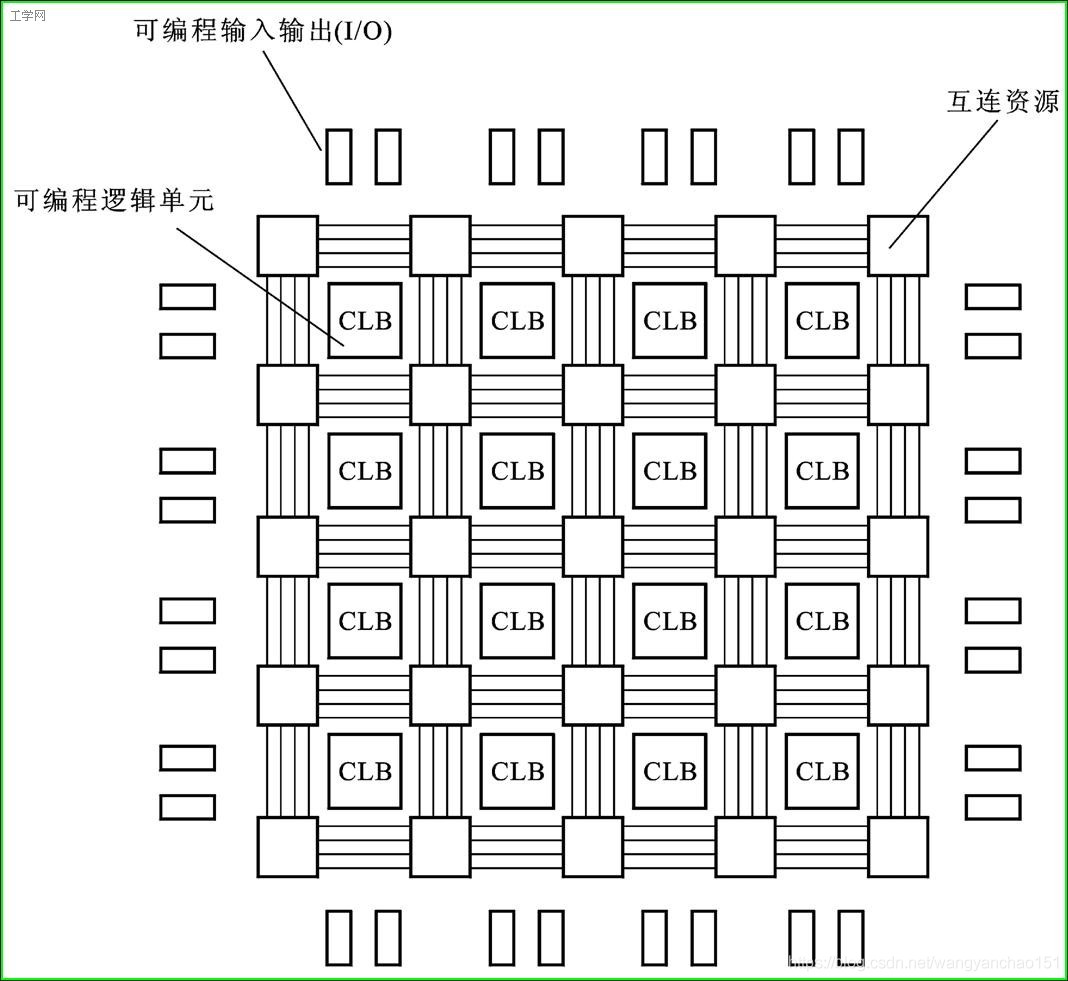
## 研究背景

随着市场环境对集成电路的功能要求愈加严格，芯片的复杂度与集成度也越来越高。目前的专用集成电路芯片 ASIC（Application Specific Integrated Circuit，专用集成电路）在目前的业界已经得到广泛的应用，且SoC已经不局限于将一个完整的系统设计集成在一块芯片上。由于 ASIC 的复杂性，使得 ASIC 的设计风险也大幅增加。一旦芯片设计失败，不仅会损失大量的研发成本，还可能导致客户对公司的产品失去信任。所以，验证逐步变为复杂芯片研发过程中的关键环节。最新一项统计数据表明，在一些项目中，芯片的设计团队规模甚至不如验证团队规模，验证所花费的时间也占到产品研发周期的一半以上，找到有效可行的验证方法成为验证的关键。目前，最常见的验证方法是软件仿真技术，但是软件仿真虽然简单易行，在面对复杂的集成电路设计时，其验证效率过低，最终可能影响到产品的及时发布。而基于 FPGA 的原型验证方法，凭借其可反复擦除、可真实的模拟设计中的硬件行为以及能够系统全面的反映设计功能等优势，被越来越多的验证团队所采用。FPGA 原型验证是指利用 FPGA 实现设计并在 FPGA 上检测设计的功能。

目前随着高性能 ASIC 芯片设计复杂度的提升以及体系结构的不断创新，其规模愈加庞大，正确性验证已成为高性能芯片研发过程中的关键环节，因此集成电路公司相继推出了许多高级的验证方法来保证验证的高效性和准确性。在复杂的高性能芯片验证方面，主要的验证方法分为三类：软件模拟验证、硬件加速器仿真验证、FPGA原型验证。在芯片功能准确性的验证过程中，软件模拟验证是最常采用的方法，其优点是模拟精确、方便灵活，因此芯片的模块功能验证一般采用软件模拟的方法。但是，随着芯片复杂度的提升，软件模拟的效率将迅速降低，并且越来越难以覆盖芯片整体的功能验证。实践表明，不论基于何种算法去设计软件工具的模拟方式，当芯片所包含的电路门数超过一定数目时，软件工具将无法准确反映系统信号的变化，验证效率受到极大制约，很难实现高效的验证效果。针对软件模拟仿真的局限性，各大集成电路公司提出了硬件加速器仿真验证，主要思想是做出专门的硬件加速器，将软件模拟中的某些复杂运算交给硬件加速器计算，最后硬件加速器将运算结果传回仿真软件，最终从整体上提升验证效率。而且硬件加速器验证的仿真环境也更加真实，相比于软件去实现芯片电路的模型构建，硬件加速器采用真实的硬件仿真环境，可以真正的反映系统设计中的各种问题，有能力检测出许多软件模拟方法很难检测的设计缺陷，这对于验证工程师的调试很有帮助，典型的硬件仿真器有 Cadence 公司的 Palladium 和 Mentor 公司的 Veloce 系列。但是硬件加速仿真验证也存在缺陷，因为这种方法需要专门的团队进行硬件加速器的设计，并且还需要设计软件模拟器与硬件加速器之间的通信接口，而一旦所验证的设计改变，原来的硬件加速器便无法再次重复利用，只能重新研发新的硬件加速器和通信接口。从理论上讲，原型验证也可以通过硬件仿真器实现，但硬件仿真器的价格太昂贵，而且由于 FPGA 的技术积累已经比较成熟，所以借用FPGA可反复擦除的特性，使用FPGA来进行原型验证成为了当下阶段较优的选择。随着 FPGA 性能的不断提升，基于 FPGA 的原型验证逐渐成为大规模复杂芯片验证的选择。



FPGA是可编程逻辑器件（PLD）的一种，是可以用来实现任意逻辑电路的集成电路。它的出现是集成电路快速发展的一个缩影。上世纪中后期，集成电路的设计成本越来越高，各大公司都希望找到一种不仅功能强大而且设计成本低廉的芯片。在这种背景下，现场可编程门阵列FPGA应运而生。FPGA大致由三部分构成：第一部分是实现逻辑电路的逻辑要素即逻辑块（Logic Block，LB）也就是我们常说的基本逻辑单元，可编程逻辑中逻辑块的实现方式有多重，比如最早的乘积项和现在的查找表、数据选择器等，但是无论哪种方式都是由可以实现任意逻辑电路的可编程部分触发器即Flip-Flop，FF等数据存储电路和数据选择器组成。逻辑要素也是在FPGA设计开发流程中比较紧俏的资源种类之一，FPGA供应商对各自的逻辑块结构有不同的称呼，如Xilinx公司称其为CLB(Configurable Logic Block)，采用的是查找表的方式，但基本原理都是类似的。第二部分是和外部进行信号输入、输出对的要素即（IO块），第三部分是连接前两种元素的布线要素既FPGA内部互联资源。

相对于其他芯片，使用FPGA进行研发的容错成本更低。在研发过程中，一旦发现芯片功能不符合所制定的要求，则马上可以对设计进行调整。在目前的研发过程中，研发团队首先会使用少量的FPGA芯片去进行设计实现，并根据实际情况进行修改直到设计达到预期的效果，最后再将达到要求的设计用ASIC实现并最终投入生产。另外，在一些功能灵活、规模较小的项目中，研发者直接使用FPGA实现设计，不仅可以满足功能需求，而且研发成本很低。

图 多FPGA仿真或验证系统流程图

对于能够使用多FPGA进行正确的验证，则必然需要进行逻辑划分，而为实现正确且合理的逻辑划分，首先要解决的就是设计逻辑综合时间长的问题。一个设计规模达到几亿门甚至数十亿门的大规模SoC设计，若使用整个设计先进行综合，再进行逻辑分割的方式，设计的综合或编译时间会非常的漫长，一次综合或编译流程就有可能花费的时间长达数天甚至数周，若设计的规模特别大，甚至会出现编译软件或服务器崩溃的情况。显然，面对超大规模的SoC设计采用RTL（Register Transfer Level，寄存器传输级别）级分割可以有效解决设计逻辑综合时间长的问题，同时实现并行编译、综合、布局布线的流程，从而有效提升FPGA原型验证的效率。但为正确实现RTL级的逻辑划分，则需要解决以下几个关键问题：

（1）首先，在对超大型SoC设计不进行逻辑综合前提下进行RTL级别分割时，需确定最合理的逻辑划分的边界，满足SoC设计的全局时钟树，同时使得进行逻辑划分后的SoC设计满足整个大型FPGA原型验证系统的互联资源和每个划分后的子设计不超出每个FPGA可综合的资源的数量（为增加FPGA综合及布局布线的成功率，常使用FPGA资源总量的60%-70%）；

（2）其次，由于无法对设计进行整体综合，则无法获得设计在FPGA上所需的准确片上资源数量，则如何对设计在RTL级进行快速资源估算，在不进行逻辑综合为前提下获得设计中每个模块的资源，从而为确认划分的边界提供参考依据将成为关键问题；

（3）最后，如何通过已经获得的RTL级设计的资源与大型原型FPGA设计平台相结合，能同时满足FPGA间的互联结构、互联资源、片上逻辑资源等条件实现合理的逻辑划分，并且划分结果与原始设计逻辑等价；

本文将主要讨论上述问题中的（1）（2），即如何对RTL设计进行快速且满足RTL级逻辑划分所需精度的快速逻辑资源估算方法。

## 国内外研究现状

现场可编程门阵列（Field Programmable Gate Array, FPGA）是一种重要的半导体器件，可通过现场重新编程以实现用户所需的逻辑设计。

当设计者使用硬件描述语言对电路进行寄存器传输级（Register Transfer Level，RTL）的编程设计后，RTL描述将经过多个设计综合步骤，转化为配置FPGA内部结构的比特流数据。这些设计综合步骤，包括逻辑综合、技术映射、逻辑打包、布局、布线等。

FPGA的硬件结构包含不同类别的资源，主要有实现逻辑电路需求的逻辑资源与存储资源，提供外部接口的输入/输出资源，以及连接前两类单元的布线资源。具体来讲，一般包括查找表（LUT）、触发器、数字处理单元（DSP）、块状存储（Block RAM，BRAM）等多种。其中，查找表和触发器等逻辑资源一般存在于逻辑块（logic block或slice）中。在设计综合流程中，逻辑综合将RTL描述转化为与FPGA结构无关的门级网表；之后技术映射把门级网表转化为由目标FPGA上不同单元组成的查找表级网表；逻辑打包再将查找表级网表中的多个查找表和触发器聚合到一个逻辑块；布局阶段为逻辑块等单元寻找FPGA上的位置；布线阶段使用布线资源，实现逻辑块等单元之间的连接；最后，生成配置FPGA具体结构的比特流。

对于给定的RTL描述，它所需的FPGA片上硬件资源数量一般需要在完成逻辑打包和布线后，才可以全部得到。但对于规模较大的电路设计，从RTL描述到布线的设计综合流程往往需要花费几个小时乃至数十小时。当出现片上资源不足的问题时，设计者需要修改RTL描述并进行耗时较长的设计综合流程迭代。

由于硬件仿真流程包含多个步骤，即RTL级分割、布局、逻辑综合、工艺映射、单个FPGA的布局和布线，由于硬件仿真流程步骤繁多且耗时较长，设计者在编写和修改RTL描述时，无法快速预知电路在FPGA上实现时需要的片上硬件资源数量，这样就可能导致无法完成技术映射或者无法完成布线，并导致多次设计迭代。另外，逻辑综合中，每一部分RTL描述所需要的实际硬件资源数量，也对RTL到门级网表的转化与优化过程有指导意义。

综上所述，如何在RTL分割阶段快速得到RTL描述需要的片上资源使用量，是一个很重要的问题。

随着FPGA设计规模的越来越大，对于片上资源估算这一环节的重要性越发明显，也是必不可少的步骤，使得FPGA资源估算这一问题也逐渐成为学者和EDA厂商的研究重点，随之而来的也是多种多样的估算方法也都各有优劣，

1）2008年Xilinx公司Paul Schumacher 和 Pradip Jha提出由于用户在对FPGA的使用过程总，总需要对设计进行逻辑综合，从而可以通过逻辑综合生成的网表结构给予现有的器件匹配库对原设计进行匹配替换，从而得到初步的资源估算结果;

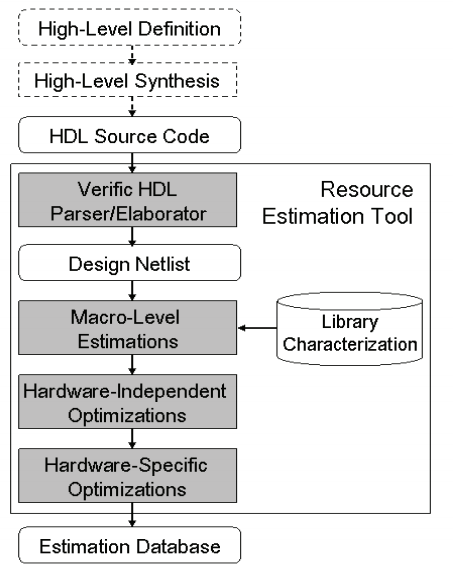
2）2011年代尔夫特工业大学Roel Meeuws等人提出可以通过深度神经学习建立基于统计方法的高级定量硬件预测模型，为人工智能在资源估计这一问题上的实现建立可行性分析;

图 一.2

3）2017年法国瓦伦西亚大学Mariem MakniLAMIH发布的论文中就提出一种快速粗略的FPGA资源计算方法，对电路设计进行快速粗略的逻辑综合、技术映射、布局布线等流程，得到的结果作为资源的估计数量；

## 研究的意义

对于给定的RTL描述，它所需的FPGA片上硬件资源数量一般需要在完成逻辑打包和布线后，才可以全部得到。但对于规模较大的电路设计，从RTL描述到布线的设计综合流程往往需要花费几个小时乃至数十小时。当出现片上资源不足的问题时，设计者需要修改RTL描述并进行耗时较长的设计综合流程迭代。

由于硬件仿真流程包含多个步骤，即RTL级分割、布局、逻辑综合、工艺映射、单个FPGA的布局和布线，由于硬件仿真流程步骤繁多且耗时较长，设计者在编写和修改RTL描述时，无法快速预知电路在FPGA上实现时需要的片上硬件资源数量，这样就可能导致无法完成技术映射或者无法完成布线，并导致多次设计迭代。另外，逻辑综合中，每一部分RTL描述所需要的实际硬件资源数量，也对RTL到门级网表的转化与优化过程有指导意义。

综上所述，如何在RTL分割阶段快速得到RTL描述需要的片上资源使用量，是一个很重要的问题。

## 主要内容及组织结构

针对我国需要开发具有自主知识产权的大型FPGA原型验证流程和相应的支持EDA软件，本文首先介绍了FPGA的基本结构及大型FPGA原型验证流程中的逻辑综合、技术映射与机器学习模型的一般方法作了详细描述，最后分别以上述基本流程为基础实现了以传统流程为基础的快速资源估算模型及以机器学习为基础的RTL级资源估算模型；两个模型的主要工作体现在以下几个方面：

（1）快速资源估算模型建模及代码实现。

（2）在快速资源估算模型的技基础上引入优化及约束项，进一步提升其精度。

（3）RTL级面向资源估算的代码特征提取设计及代码实现。

（4）基于机器学习的RTL级资源估算模型训练及验证。