

TURING

图灵程序
设计丛书

FPGA

原理和结构

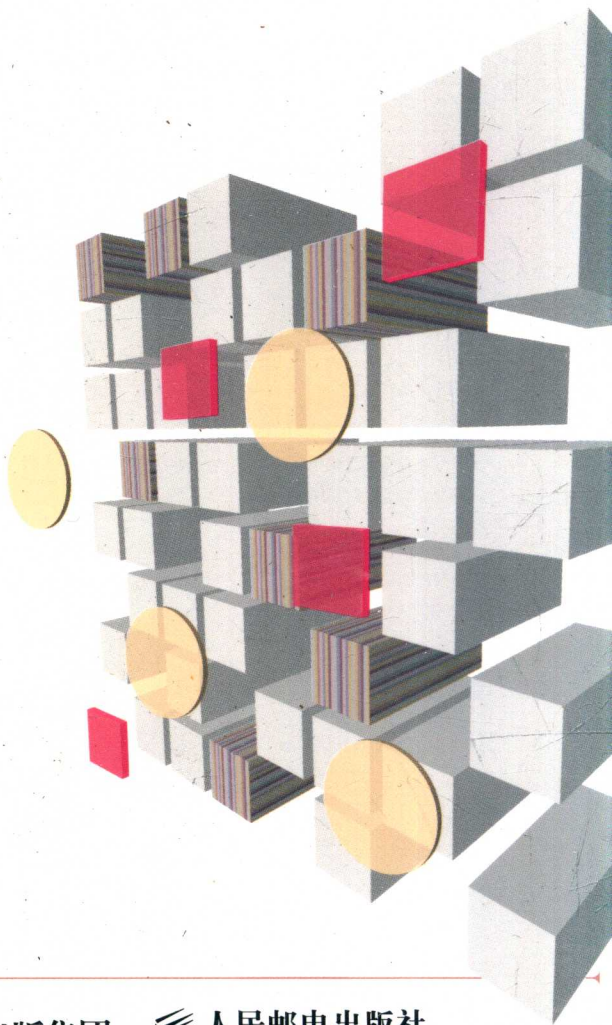
[日] 天野英晴 —— 主编

赵谦 —— 译

日本可重构计算
专家团队撰写!

一本书了解
FPGA技术及应用
基本原理

简明深入 夯实基础
FPGA领域入门佳作



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

FPGA

原理和结构

[日] 天野英晴 — 主编

赵谦 — 译

人民邮电出版社
北京

图书在版编目(CIP)数据

FPGA原理和结构/(日)天野英晴主编;赵谦译

北京:人民邮电出版社,2019.3

(图灵程序设计丛书)

ISBN 978-7-115-50331-2

I. ①F… II. ①天… ②赵… III. ①可编程序逻辑器件—系统设计 IV. ①TP332.1

中国版本图书馆CIP数据核字(2019)第007899号

内 容 提 要

本书由日本可重构领域权威教授团队撰写,是一本讲解FPGA原理的书。前5章从FPGA的相关概念入手,简明又严谨地阐述了FPGA硬件构成和CAD工具的内部原理等理论基础,有助于读者快速入门,也有助于读者了解如何优化自己的电路代码,获得更高的性能。第6章分析了FPGA的优势,系统地介绍适用于FPGA开发的“硬件算法”,有助于读者利用FPGA更高效、更有针对性地解决问题。第7章和第8章则讲解了FPGA的技术动态和应用案例。

本书适合所有对FPGA感兴趣,特别是想了解FPGA原理的工程师阅读,也可作为各大院校相关专业师生的参考指南使用。

◆主 编 [日]天野英晴

译 赵 谦

责任编辑 高宇涵

责任印制 周昇亮

◆人民邮电出版社出版发行 北京市丰台区成寿寺路11号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

大厂聚鑫印刷有限责任公司印刷

◆开本:880×1230 1/32

印张:8.875

字数:295千字

2019年3月第1版

印数:1-3 500册

2019年3月河北第1次印刷

著作权合同登记号 图字:01-2017-7989号

定价:59.00元

读者服务热线:(010)51095186转600 印装质量热线:(010)81055316

反盗版热线:(010)81055315

广告经营许可证:京东工商广登字20170147号

天野英晴

1986年于日本庆应义塾大学获得博士学位。现任日本庆应义塾大学教授。主要从事高性能计算和可重构系统研究。日本可重构系统研究会发起人，国际学会“高效加速器及可重构技术”（HEART）创立者之一。

赵谦

2014年于日本熊本大学获得博士学位。现任日本九州工业大学助理教授。主要从事可重构系统的架构、设计方法及应用研究。

站在巨人的肩上
Standing on Shoulders of Giants



iTuring.cn

版 权 声 明

Original Japanese Language edition

FPGA NO GENRI TO KOSEI

by Hideharu Amano

Copyright © Hideharu Amano 2016

Published by Ohmsha, Ltd.

Chinese translation rights in simplified characters arranged with Ohmsha, Ltd.
through Japan UNI Agency, Inc., Tokyo

本书中文简体字版由 Ohmsha, Ltd. 授权人民邮电出版社独家出版。
未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

《FPGA原理和结构》

主编、执笔人一览

主 编	天野英晴 (庆应义塾大学)	
执笔人	末吉敏则 (熊本大学)	(第1章)
	饭田全广 (熊本大学)	(第2章, 第5章)
	柴田裕一郎 (长崎大学)	(第3章)
	尼崎太树 (熊本大学)	(第3章)
	密山幸男 (高知工科大学)	(第4章)
	泉知论 (立命馆大学)	(第4章)
	中原启贵 (东京工业大学)	(第6章)
	佐野健太郎 (东北大学)	(第6章)
	长名保范 (琉球大学)	(第7章)
	丸山勉 (筑波大学)	(第7章)
	山口佳树 (筑波大学)	(第7章)
	张山昌论 (东北大学)	(第8章)
	本村真人 (北海道大学)	(第8章)
	渡边实 (静冈大学)	(第8章)

译者序

近些年，CPU 等通用处理器的性能提升速度放缓，为了继续满足各行各业对高能效计算日益增长的需求，FPGA 作为可重构计算体系的代表器件，一夜之间在众多新兴热点领域受到广泛关注。

我在旅日留学和工作的 10 年间，亲身经历了 FPGA 技术焕发青春的这一过程。10 年前我初到日本求学时，FPGA 主要用作半导体行业的基础工具。在学校，学生通过 FPGA 学习电路编写、电路分析、硬件算法等芯片开发技术；在半导体企业中，工程师们使用 FPGA 对芯片进行仿真和验证，几乎很少在最终产品中看到 FPGA 的身影。而如今，人工智能、大数据分析、网络通信、图像处理、机器人等众多领域的教学和研发过程都引入了 FPGA，FPGA 相关的学术会议也逐渐汇集了各个领域的专家学者，彼此交流和切磋 FPGA 在各自领域的应用成果。可以说，FPGA 已经成为一种主流的通用计算技术。

在云计算领域，已经有众多研究和产业应用证明了 FPGA 技术可以有效提高各种云端负载的处理性能，同时还可以降低功耗。微软公司于 2014 年率先公布了运用 FPGA 将搜索引擎 Bing 的性能提升一倍的成功案例，这被认为是 FPGA 技术进入云计算领域的一个里程碑事件。随后，亚马逊公司于 2016 年年末在其 EC2 云计算平台中首次加入了 FPGA 实例，这让任何人都能以相对低廉的价格按需租用 FPGA，大大降低了 FPGA 的使用门槛。可喜的是，国内互联网厂商在 FPGA 应用上并不落后于国外互联网巨头，大家很早就认识到了 FPGA 在云端应用上的巨大潜力，并积极投身其中。就在亚马逊公司公布其云端 FPGA 实例后不久，百度云、阿里云、腾讯云、华为云也都迅速上线了云端 FPGA 实例并开启公测。在云端 FPGA 起步后的短短一年多时间里，各大互联网厂商已经在人工智能、基因测序、图像处理、视频转码、大数据分析、数据库等关键领域取得了令人瞩目的应用成果，FPGA 正在这些云端应用

中为数十亿用户提供高性能、低功耗的计算保障。

在边缘计算领域，FPGA 也同样应用广泛。边缘计算是随着物联网的普及而越来越受关注的一种计算形态，它在靠近物或数据源头的一侧就地提供计算、存储、网络等服务。通常，边缘计算在追求较高数据处理能力的同时，还要求处理器具备小型化、低功耗和灵活应对各种应用场景的能力。而近些年流行起来的 SoC FPGA，通过在单个芯片上集成 ARM 硬核处理器和 FPGA，充分迎合了上述几点要求。至今，我们已经看到 SoC FPGA 产品被广泛应用在智能摄像头、自动驾驶、无人机、摄像机、智能语音助手等家喻户晓的电子产品当中。

FPGA 应用领域的扩张速度非常快，以至于市面上还没有一本教科书能够囊括其全貌。为了将 FPGA 的历史、基本原理、开发方式以及最新的应用动向通俗易懂地传递给更多的读者，日本可重构系统研究会的老师们携手编写了本书。如今，在日本学习和使用 FPGA 的人群当中，本书几乎是人手一册。我非常荣幸受本书主编天野英晴教授、我的恩师末吉敏则教授和饭田全广教授所托，将这本 FPGA 领域的优秀教材带给国内的读者。

本书的编者和作者们可谓是日本可重构计算领域的全明星阵容，他们数十年如一日工作在可重构计算的教育和科研前沿，为日本乃至世界的可重构计算做出了令人瞩目的贡献。本书是一本 FPGA 技术的综述图书，旨在向读者传递 FPGA 技术及其应用的基本原理。作者们在各个章节深入浅出、旁征博引，以多年的专业经验提炼概要，让读者能在尽量短的时间内了解 FPGA 领域的全貌。而希望在某个领域深入研究的读者，通过参阅各章的参考文献也能事半功倍。

IT 行业正处在一个深刻变革的时期，沿用数十年的 CPU 架构已然不能满足全球爆炸式增长的计算需求，以软硬协同开发为特征的计算机体系结构的黄金时代正在到来。FPGA 作为这场技术变革的主角，未来一定会在更多领域大展拳脚。当硬件变“软”，我们对硬件、软件、计算机系统的认知又将如何发展呢？衷心希望读者能在阅读本书的过程有所收获，如果能进一步激发读者对科技的发展产生独到的见解，相信对编写本书及推进本书出版的全体人员都是最大的鼓舞。

赵 谦

2018 年 6 月 11 日

序

近些年，以 FPGA（Field Programmable Gate Array，现场可编程门阵列）为代表的可编程芯片的发展十分活跃，这些芯片被广泛应用在智能手机、平板电脑、汽车、数字电视、音响、网络设备、计算机以及几乎所有的家电产品之中。可编程芯片一直在世界范围内引领尖端制程技术向前发展，现如今已经可以在一枚芯片上实现具有存储器、处理器、各种输入/输出等模块的大规模系统。

令人感叹的是，FPGA 作为应用如此广泛的电子元器件，其知名度却不高。提到微处理器 CPU 和系统 LSI，即便是门外汉也知道它们是先进的电子器件。而提到 FPGA，不但普罗大众闻所未闻，即使是一些工科专业的学生也所知甚少。这主要是因为早期的 FPGA 比系统 LSI 速度慢、功耗大、成本高，所以没有受到日本厂商的重视。而且，早期的 FPGA 内部构造单纯，设计技术也相对简单，通常只会被当作 LSI 芯片设计的子科目来对待。因此在过去的教学当中，FPGA 仅出现在数字电路课本的最后几页，在大学通常也只会数字电路课程的末尾或是在学生实验中简单提及一下就结束了。

然而现在，FPGA 已经超越标准数字芯片和系统 LSI，成为构建数字系统的主角。可以说，理解 FPGA 的结构和特征并熟练使用 FPGA，是开发 IT 产品的必备知识和技能。FPGA 器件的技术体系自成一脉，它的结构、设计方法、设计环境和应用方法与通常的 LSI 设计都不同。因此，如果缺乏这些知识背景，即便是 LSI 设计专家也难以将 FPGA 的功能发挥到极致。

本书将全面介绍 FPGA 与可编程芯片相关的历史、结构、架构技术、设计环境、设计方法、硬件算法、输入/输出、应用系统和新型器件等知识。执笔本书的作者们都是日本电子信息通信学会下属可重构系统研究会的主要成员，他们都对该领域最前沿的知识了如指掌。只读这

一本书，读者就可以了解可重构系统的全貌。

本书以大学、高中、高专的学生为主要目标读者，从数字系统基础、可编程逻辑基础开始讲起。同时，本书还包含了领域内的前沿内容，同样适合正在使用 FPGA 的技术从业者阅读。

第 1 章介绍阅读本书所需掌握的基础知识。在这一章的结尾部分总结了 FPGA 的发展史，即使是非常了解该领域的读者，也推荐读一读此部分。第 2 章介绍 FPGA 的概要，刚入门的读者只要阅读前半部分就基本可以理解什么是 FPGA 和可编程芯片技术了，而后半部分则包含了一些进阶内容。第 3 章对 FPGA 的结构进行深入讲解，推荐 FPGA 领域的专业读者阅读。以上 3 章就是介绍 FPGA 基础知识和构造的部分。

第 4 章介绍 FPGA 的设计流程和工具，其中还会包含最新的高层次综合技术。这一章的内容适合刚入门的 FPGA 开发者。第 5 章将深入介绍用于设计 FPGA 的 CAD 工具内部原理。这部分内容面向 FPGA 专业读者，阅读本章读者可以了解到用于设计 FPGA 的 CAD 工具不同于其他 LSI 设计工具的独到之处。

第 6 章介绍的是可以应用在 FPGA 上的硬件算法。如果想要充分发挥 FPGA 的实力，那么采用高效的硬件算法是十分必要的。第 7 章介绍 FPGA 的应用实例涉及金融、大数据、生化和宇宙开发，读者可以从中看到 FPGA 技术所蕴含的巨大的可能性。第 8 章介绍一些 FPGA 相关的最新技术动向，比如 FPGA 的输入/输出接口、光通信、粗粒度可重构系统、异步 FPGA 等。

FPGA 初学者通过阅读第 1 章、第 2 章前半部分和第 4 章就可以掌握相关基础知识。具有一定经验的 FPGA 专业读者通过阅读第 3 章和第 5 章可以加深对 FPGA 技术的理解，阅读第 6 章到第 8 章则会对今后的设计工作和产品开发有所帮助。

像 FPGA 这样的可编程芯片可以说是电子设计爱好者梦寐以求的一种器件：无须焊接，只要几百日元（几十元人民币）的芯片配合免费的开发工具就能实现性能惊艳的数字系统。希望通过阅读本书，有更多的读者可以感受到 FPGA 这种卓越技术的魅力。

主编 天野英晴

2016 年 3 月

目 录

第 1 章 理解 FPGA 所需的基础知识

1.1 逻辑电路基础	1
1.2 同步电路设计	6
1.3 FPGA 的定位和历史	11
1.4 FPGA 专业术语	27
参考文献	33

第 2 章 FPGA 的概要

2.1 FPGA 的构成要素	36
2.2 可编程技术	38
2.3 FPGA 的逻辑实现	47
参考文献	59

第 3 章 FPGA 的结构

3.1 逻辑块的结构	61
3.2 逻辑簇	66
3.3 自适应查找表	67
3.4 布线线段	71
3.5 开关块	77
3.6 连接块	80
3.7 I/O 块	81
3.8 DSP 块	85
3.9 硬宏	88

3.10 嵌入式存储器	89
3.11 配置链	92
3.12 PLL 和 DLL	94
3.13 典型的 PLL 块	95
3.14 PLL 块的自由度和限制	96
参考文献	98

第 4 章 设计流程和工具

4.1 设计流程	105
4.2 基于 HDL 的设计流程	106
4.3 HLS 设计	116
4.4 基于 IP 的设计方法	126
4.5 包含处理器的设计	128
参考文献	132

第 5 章 设计原理

5.1 FPGA 设计流程	134
5.2 工艺映射	135
5.3 逻辑打包	139
5.4 布局布线	144
5.5 低功耗设计工具	149
参考文献	153

第 6 章 硬件算法

6.1 流水线结构	156
6.2 并行计算和 Flynn 分类	160
6.3 脉动算法	162
6.4 数据流机	169
6.5 流处理	177

6.6 细胞自动机	182
6.7 硬件排序算法	184
6.8 模式匹配	185
参考文献	194

第 7 章 PLD/FPGA 应用案例

7.1 可编程逻辑器件的现在和未来	198
7.2 超级计算机：大规模系统中的 PLD/FPGA	199
7.3 网络通信领域：实现高速、高带宽通信的 PLD/FPGA	204
7.4 大数据处理：Web 搜索	212
7.5 基因科学：短序列拼接	214
7.6 金融市场：FPGA 创造巨大财富	216
7.7 人工智能：在 FPGA 上实现深度学习之后	220
7.8 图像处理：搜索太空垃圾	223
参考文献	225

第 8 章 新器件与新架构

8.1 粗粒度可重构架构	237
8.2 动态重配置架构	239
8.3 异步 FPGA	244
8.4 FPGA 系统的低功耗化技术	251
8.5 3D-FPGA	257
8.6 高速串行 I/O	259
8.7 光可编程架构	263
参考文献	266

第 1 章

理解 FPGA 所需的基础知识

1.1 逻辑电路基础

FPGA (Field Programmable Gate Array, 现场可编程门阵列) 是一种可通过重新编程来实现用户所需逻辑电路的半导体器件。为了便于大家理解 FPGA 的设计和结构, 我们先来简要介绍一些逻辑电路的基础知识^[1-3]。

1.1.1 逻辑代数

在逻辑代数中, 所有变量的值只能取 0 或者 1。逻辑代数是而与逻辑值 (0 和 1) 相关的逻辑与 (AND)、逻辑或 (OR) 和逻辑非 (NOT) 三种运算形成的代数体系, 也称为布尔代数。

表 1-1 列出了定义逻辑与、逻辑或的二元运算以及定义逻辑非的一元运算。在这里, 三种运算分别使用 “ \cdot ” “ $+$ ” 和 “ $\bar{\quad}$ ” 运算符号来表示。逻辑与 $x \cdot y$ 是指 x 和 y 都为 1 时, 结果为 1 的运算。逻辑或 $x + y$ 是指 x 或 y 至少有一方为 1 时, 结果为 1 的运算。逻辑非 “ \bar{x} ” 是取相反逻辑值的一元运算: 如果 x 为 0 则结果为 1; 反之, 如果 x 为 1 则结果为 0。

表 1-1 逻辑运算 (布尔代数的公理)

逻辑与 (\cdot)	逻辑或 ($+$)	逻辑非 ($\bar{\quad}$)
$0 \cdot 0 = 0$	$0 + 0 = 0$	$\bar{0} = 1$
$0 \cdot 1 = 0$	$0 + 1 = 1$	
$1 \cdot 0 = 0$	$1 + 0 = 1$	$\bar{1} = 0$
$1 \cdot 1 = 1$	$1 + 1 = 1$	

逻辑代数满足表 1-2 所示的定理。这里的符号“=”表示其两边的计算结果总是相等，即等价。如果对换逻辑表达式中的逻辑值 0 和 1、逻辑运算“与”和“或”，对换后得到的新逻辑表达式与对换前的表达式运算顺序不变，那么新逻辑表达式就称为原逻辑表达式的对偶式。逻辑代数中，如果某定理的逻辑表达式成立，其对偶式也成立。

表 1-2 布尔代数的定理

零元 $x \cdot 0 = 0, x + 1 = 1$	单位元 $x \cdot 1 = x, x + 0 = x$
幂等律 $x \cdot x = x, x + x = x$	补余律 $x \cdot \bar{x} = 0, x + \bar{x} = 1$
双重否定 $\overline{\overline{x}} = x$	交换律 $x \cdot y = y \cdot x, x + y = y + x$
结合律 $(x \cdot y) \cdot z = x \cdot (y \cdot z), (x + y) + z = x + (y + z)$	
分配律 $x \cdot (y + z) = (x \cdot y) + (x \cdot z), x + (y \cdot z) = (x + y) \cdot (x + z)$	
吸收律 $x + (x \cdot y) = x, x \cdot (x + y) = x$	
德摩根定律 (De Morgan's laws) $\overline{x + y} = \bar{x} \cdot \bar{y}, \overline{x \cdot y} = \bar{x} + \bar{y}$	

1.1.2 逻辑表达式

逻辑表达式是用来描述运算过程的算式，由逻辑运算符、任意数量的逻辑变量以及必要的括号和常数值 0 或 1 组合而成。对于包含 n 个逻辑变量 $x_1, x_2, x_3, \dots, x_n$ 的逻辑表达式来说，我们先在其各个逻辑变量内代入逻辑值 0 或 1，形成任意组合（共 2^n 组），然后依照逻辑表达式的计算步骤计算这些组合，就可以得到值为 0 或 1 的计算结果。也就是说，逻辑表达式定义了具有某种逻辑功能的逻辑函数 $F(x_1, x_2, x_3, \dots, x_n)$ 。在没有括号的情况下，逻辑与的计算优先于逻辑或。逻辑与的运算符“ \cdot ”也可省略。

任何逻辑函数都可以由逻辑表达式来描述，而且描述同一逻辑函数的逻辑表达式可以有多个。逻辑表达式的标准形式指的是通过增加表达式形式上的限制，使得一个逻辑函数只有一个逻辑表达式与之对应的情况。

逻辑表达式中，逻辑变量以原变量或反变量的形式出现。原变量和反变量统称为字面量（literal）。字面量的逻辑与（每个字面量不能出现多次）叫作与项，与项的逻辑或运算叫作积之和。包含所有字面量的与项称为最小项，由最小项构成的积之和称为标准积之和（标准积）。

将标准积的逻辑与和逻辑或对调即为标准和之积。字面量的逻辑或（每个字面量不能出现多次）叫作或项，或项的逻辑与运算叫作和之积。包含所有逻辑变量的或项称为最大项，由最大项构成的和之积称为标准和之积（标准和）。

1.1.3 真值表

除了逻辑表达式，逻辑函数的描述方法还包括真值表和逻辑门。针对逻辑函数所有可能的输入组合——列出输出值，我们就可以得到真值表。对于组合逻辑电路，只要列出所有可能的输入和对应的输出值，就可以完整地描述电路功能。因此，电路功能通常使用真值表描述。输入的个数为 n 时，真值表的组合数为 2^n 。真值表中，需要记入每组输入值所对应的输出值。

描述逻辑函数的逻辑表达式可以有多个，而描述逻辑函数的真值表却是唯一的。虽然一个逻辑表达式只描述一个逻辑函数，但一个逻辑函数可以通过无数的等价逻辑表达式来描述。实现真值表所定义的功能的电路称为查找表（Look-up Table, LUT），是当前主流FPGA的基本单元。

从真值表推导逻辑表达式的形式有两种：“积之和表达式”与“和之积表达式”。在真值表输出为1的行中取输入变量的与项（最小项），然后将这些最小项相或，即可得到标准积之和表达式；相对地，在真值表输出为0的行中取输入变量的反变量的或项（最大项），然后将这些最大项相与，即可导出标准和之积表达式。图1-1中的示例展示了如何从真值表推导逻辑表达式。