

# デジタル回路設計入門

— FPGA時代の論理回路設計 —

博士(工学) 中野 浩嗣  
博士(工学) 伊藤 靖朗

【共著】

コロナ社

# まえがき

本書は、デジタル回路設計を初めて学ぶ大学の学部生を対象とした教科書である。前提となる知識は高校の数学 I 程度のレベルであり、高校生が読んで理解できるように書かれている。

ほかの一般的なデジタル回路設計や論理設計の教科書と違い、プログラマブル集積回路である FPGA の設計を意識した内容になっており、実際のデジタル回路設計の際に考慮しなければならない事柄について、詳しく説明している。また、理論的な回路計算量の考え方も含んでいる。すでにデジタル回路設計を学んだ大学院生や社会人が読んで、デジタル回路設計に関する新たな知識を得られるようになっている。実際にデジタル回路設計を行っている技術者が経験によって獲得し、意識せずに用いているノウハウ的な内容も含んでおり、そのような技術者にとっても本書を読むことは有益である。

本書は 6 章で構成される。

1 章は、論理式と、基本ゲート回路を用いた組み合わせ回路について説明している。カルノー図やクワイン・マクスキー法を用いた論理式の単純化がおもな内容である。論理関数の表現方法として、真理値表、論理式、組み合わせ回路の 3 通りがあるが、これらが相互に変換可能であることを説明している。これらの内容に加えて、汎 2 入力ゲートとそれを用いた組み合わせ回路の大きさの評価法について述べている。これは回路計算量を理論的に議論するベースとなる。

2 章は、よく使われるさまざまな組み合わせ回路について紹介している。特に、実際の回路でよく用いられるセレクト回路とデコーダ回路の設計方法について詳しく説明している。これらに加えて、セレクト回路と汎 2 入力ゲートを用いた任意の  $n$  ビット論理関数を計算する回路の構成法を説明している。こ

れは他書にない特徴的な内容で、回路計算量理論の基礎となる考え方を示している。

3章は、整数や小数点数のビット列による表現方法と、加算回路、減算回路、乗算回路の設計方法について説明している。特に、オーバーフローの判定法や2の補数の場合の乗算回路についても紹介している。

4章は、ラッチやフリップフロップ等の記憶回路を基本ゲート回路を用いて構成する方法について紹介している。また、フリップフロップのアクセスタイム、セットアップタイム、ホールドタイム等の時間定数について説明しており、フリップフロップが正しく動作するための条件を示している。また、フリップフロップを用いた最も簡単な回路の例として、非同期カウンタ回路を説明している。

5章は、フリップフロップと組み合わせ回路を用いた順序回路の設計法である。まず、 $2^n$ 進カウンタの設計法と、最大動作周波数の求め方を説明している。続いて、2のべきでないカウンタの例として5進カウンタの設計法を示しているが、多くの人により共通して行われている間違っただ設計法を紹介し、正しく動作しない理由を説明している。そして、改めて正しい5進カウンタの設計法を示している。FPGA等によく用いられる順序回路である完全同期式順序回路をミラー型、ムーア型、単純型の3通りに分類し、それらの特徴について解説している。また、完全同期式順序回路のレジスタトランスファレベルシミュレーションとゲートレベルシミュレーションについて解説し、正しく動作するための条件を示している。具体的な完全同期式順序回路の例として、ステートマシンとメモリ回路を紹介している。

6章は、いくつかの実用的な回路を紹介している。数取器回路を題材に、外部からの入力信号を受け付ける順序回路の設計時に注意すべき点について説明している。また、FPGAに組み込まれているルックアップテーブル回路とその使い方について述べている。最後に畳み込み回路の設計と、パイプライン化による高速化手法、およびFPGAが組み込んでいるDSP回路を用いた畳み込み回路の実装方法について紹介している。

以上により、実践的なデジタル回路設計や FPGA の設計を行うにあたり、基礎的な内容をマスターでき、また、回路計算量の理論的な分析も行えるようになる。

最後に、本書を出版するにあたり、コロナ社に深く感謝を申し上げる。

2021 年 2 月

中野浩嗣，伊藤靖朗

# 目 次

## 1 章 論理式と組み合わせ回路

1.1 論理式, 組み合わせ回路, および論理関数	2
1.1.1 論理演算の基本法則	5
1.1.2 基本ゲート回路と組み合わせ回路	7
1.1.3 基本ゲート回路の拡張と排他的論理和	10
1.1.4 汎 2 入力ゲートと組み合わせ回路の大きさ	14
1.2 論理式と組み合わせ回路の簡単化	17
1.2.1 論理式の標準形	17
1.2.2 論理演算の基本法則を用いた論理式の簡単化	22
1.2.3 カルノー 図	25
1.2.4 カルノー図を用いた論理式の簡単化	27
1.2.5 禁止入力付き論理関数とカルノー図	31
1.2.6 7 セグメントデコーダ回路	33
1.2.7 クワイン・マクラスキー法による論理式の簡単化	35
1.2.8 5 変数以上のカルノー図	38
1.2.9 NAND ゲートだけ, もしくは NOR ゲートだけを用いた組み合わせ回路	39
演 習 問 題	42

## 2 章 さまざまな組み合わせ回路

2.1 2 進 数	45
2.2 スリーステートバッファ回路とバス	46
2.3 セレクタ回路	48
2.3.1 2 入力セレクタ回路	48

2.3.2	多入力セクタ回路	50
2.3.3	多入力セクタ回路の再帰的設計	51
2.3.4	セクタ回路を用いた任意の論理関数計算回路	53
2.3.5	セクタ回路による論理関数計算回路の大きさ削減	56
2.4	デコーダ回路	59
2.4.1	積和形の論理式によるデコーダ回路	59
2.4.2	再帰的設計によるデコーダ回路	60
2.4.3	デコーダ回路を用いたセクタ回路	64
2.5	バレルシフト回路	65
	演習問題	67

## 3章 算術演算回路

3.1	符号付き2進数	70
3.1.1	符号付き絶対値表現	70
3.1.2	1の補数表現	71
3.1.3	2の補数表現	72
3.1.4	ビット拡張	73
3.2	インクリメント回路	74
3.3	加算回路	77
3.3.1	符号なし2進数の加算回路	77
3.3.2	2の補数の加算回路	81
3.4	減算回路	82
3.5	乗算回路	86
3.5.1	符号なし2進数の乗算回路	86
3.5.2	2の補数の乗算回路	88
3.6	小数点数	92
3.6.1	固定小数点数	92
3.6.2	固定小数点数の算術演算回路	94
3.6.3	浮動小数点数	95

演習問題	96
------	----

## 4章 記憶回路

4.1 フリップフロップの構造	99
4.1.1 RS型フリップフロップ	99
4.1.2 D型ラッチ	102
4.1.3 D型フリップフロップ	104
4.1.4 非同期セット・非同期リセット付きD型フリップフロップ	107
4.1.5 $n$ ビットフリップフロップ(レジスタ)	109
4.1.6 イネーブル付きフリップフロップ	110
4.2 フリップフロップの時間定数	112
4.2.1 セットアップタイム・ホールドタイムとメタステーブル状態	112
4.2.2 最小アクセスタイム・最大アクセスタイム	113
4.3 フリップフロップを用いた回路	114
4.3.1 T型フリップフロップ	114
4.3.2 非同期カウンタ回路	115
演習問題	117

## 5章 順序回路

5.1 カウンタ回路	121
5.1.1 カウンタ回路の設計	121
5.1.2 レジスタトランスファレベルシミュレーション	122
5.1.3 ゲートレベルシミュレーション	124
5.1.4 クロックスキュー	126
5.1.5 カウンタ回路の動作周波数	127
5.2 2のべきでないカウンタ回路	129
5.2.1 間違った設計方法	129
5.2.2 正しい設計方法	131
5.3 完全同期式順序回路	134

5.4	ステートマシンとワンホットステートマシン	138
5.4.1	ムーア型完全同期式順序回路によるステートマシンの設計	138
5.4.2	ステートマシンのハザード	143
5.4.3	単純型完全同期式順序回路によるステートマシンの設計	144
5.4.4	ワンホットステートマシンの設計	146
5.5	メモリ回路	148
5.5.1	メモリ回路の仕様	148
5.5.2	メモリ回路の設計	149
	演習問題	151

## 6章 実用的な回路

6.1	積分回路と微分回路	155
6.1.1	積分回路	155
6.1.2	微分回路と立ち上がり検出回路	156
6.2	数取器回路	158
6.2.1	立ち上がり検出回路を用いた数取器回路	158
6.2.2	メタステーブル対策を行った数取器回路	160
6.2.3	チャタリング除去回路	161
6.3	ルックアップテーブル回路	163
6.3.1	5LUT2 による 2 ビット加算器	164
6.3.2	6LUT1 による 4 入力セクタ回路	165
6.3.3	6LUT1 を用いた論理関数計算回路	166
6.4	畳み込み計算回路	167
6.4.1	畳み込み計算回路の設計	167
6.4.2	畳み込み計算回路の性能評価	170
6.4.3	組み合わせ回路のパイプライン化による高速化	171
6.4.4	DSP 回路を用いた畳み込み計算回路	173
	演習問題	175

索 引	176
-----	-----



# 1 章

## 論理式と組み合わせ回路

### ◆本章のテーマ

真理値を対象とする論理関数を表すのに、真理値表、論理式、組み合わせ回路の三つの表現方法がある。これらの取り扱いと、相互の変換方法が本章の主たるテーマである。特に、真理値表の形で論理関数が与えられたとき、その論理関数を計算する論理式の求め方とその論理式の簡単化の二つの方法（カルノー図を用いる方法とクワイン・マクラスキー法）を説明する。

### ◆本章の構成（キーワード）

#### 1.1 論理式、組み合わせ回路、および論理関数

論理演算（論理積、論理和、論理否定等）、基本ゲート回路（AND ゲート、OR ゲート、NOT ゲート等）、真理値表、汎 2 入力ゲート、組み合わせ回路の大きさ

#### 1.2 論理式と組み合わせ回路の簡単化

主加法標準形、主乗法標準形、カルノー図、禁止入力、クワイン・マクラスキー法

### ◆本章を学ぶと以下の内容をマスターできます

- ☞ 論理演算の基本法則
- ☞ 論理関数の真理値表から標準形の論理式を求め、基本ゲート回路による組み合わせ回路を設計する方法
- ☞ 真理値表の禁止入力（ドントケア入力）の意味と取り扱い
- ☞ カルノー図やクワイン・マクラスキー法を用いて最も簡単な論理式を求め、組み合わせ回路を設計する方法
- ☞ 汎 2 入力ゲートを用いた組み合わせ回路の大きさを評価する方法
- ☞ NAND ゲートのみ、もしくは NOR ゲートのみを用いて、組み合わせ回路を設計する方法

## 1.1 論理式, 組み合わせ回路, および論理関数

普通の数式は整数や実数等を対象とするが、論理式は、真と偽のいずれかの値をとる真理値（論理値とも呼ばれる）を対象とする。簡便のため、真を1、偽を0と表す。この真理値に対するいくつかの演算（論理演算）を定義する。

論理否定： $\bar{A}$  は、 $A$  が0のとき1であり、1のとき0の値をとる。つまり、 $\bar{A} = 1 - A$  である。

論理積： $A \cdot B$  は、 $A$  と  $B$  の両方が1のときのみ1となり、それ以外は0となる。よって整数に対する乗算と同じである。

論理和： $A + B$  は、 $A$  と  $B$  の両方が0のときのみ0となり、それ以外は1となる。整数の加算と似ているが、 $1 + 1 = 1$  となる点異なる。

表 1.1 は、真理値の組み合わせとその演算結果を表にした真理値表である。

表 1.1 論理否定, 論理積, および論理和の真理値表

(a) 論理否定		(b) 論理積, 論理和			
$A$	$\bar{A}$	$A$	$B$	$A \cdot B$	$A + B$
0	1	0	0	0	0
0	1	0	1	0	1
1	0	1	0	0	1
1	0	1	1	1	1

普通の数式が数と四則演算等演算の組み合わせで表されるのと同様に、真理値と論理演算の組み合わせで、論理式が表される。例えば、つぎは論理式の例である。

$$\overline{1 \cdot 0} + 0$$

数式の計算と同様に演算を順に行うことにより、この論理式の真理値は、つぎの計算により求めることができる。

$$\overline{1 \cdot 0} + 0 = \overline{0} + 0 = 1 + 0 = 1 \quad (1.1)$$

数式では乗算の優先順位が加算より高く、並んだ乗算と加算では、乗算の計算が先に行われる。同様に、論理積の優先順位は論理和より高いものとする。必要であれば、括弧を用いて計算順序を指定する。例えば、つぎのように括弧内が優先された論理式の計算が行われる。

$$0 + 1 \cdot (1 + 0) = 0 + 1 \cdot 1 = 0 + 1 = 1 \quad (1.2)$$

括弧内が最優先で計算され、また、論理積と論理和が並んでいる場合、論理積の計算が先に行われる。

数式に変数を含めることができるのと同様に、論理式に論理変数（または単に変数と呼ぶ）を含めることができる。つぎの例は、三つの論理変数  $A$ ,  $B$ , および  $C$  を含んだ論理式である。

$$\overline{A \cdot B} + C$$

三つの論理変数に対して、割り当てられる真理値が決まれば、論理式の真理値を計算することができる。例えば、 $ABC = 100$ （つまり、 $A = 1$ ,  $B = 0$ ,  $C = 0$ ）のとき、この論理式はつぎのように計算され、真理値は 1 となる。

$$\overline{A \cdot B} + C = \overline{1 \cdot 0} + 0 = \overline{0} + 0 = 1 + 0 = 1 \quad (1.3)$$

変数を含んだ数式によって関数を定義することができる。例えば、 $f(x, y) = x^2 + 2y$  は変数  $x$  と  $y$  の実数値が決まれば、実数の関数値が一意に求められる関数である。同様に、論理変数を含んだ論理式によって、論理関数を定義することができる。例えば、三つの論理変数  $A$ ,  $B$ ,  $C$  を含むつぎの二つの論理式は、論理関数  $f$  と  $g$  をそれぞれ定義している。

$$f(A, B, C) = \overline{A \cdot B} + C \quad (1.4)$$

$$g(A, B, C) = \overline{A \cdot B \cdot C} \quad (1.5)$$

#### 4 1. 論理式と組み合わせ回路

三つの論理変数  $A$ ,  $B$ , および  $C$  のとる真理値の組み合わせは,  $2^3 = 8$  通りである。このすべての組み合わせについて, 論理関数  $f$  と  $g$  がとる値は, 表 1.2 の真理値表で表すことができる。

表 1.2 論理関数  $f$  と  $g$  の真理値表

$A$	$B$	$C$	$f(A, B, C)$	$g(A, B, C)$
0	0	0	1	1
0	0	1	1	1
0	1	0	1	1
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	1	1

表より, 三つの論理関数  $f$  と  $g$  が, 論理変数の同じ値の組み合わせに対して同じ真理値をとることがわかる。つまり,  $f$  と  $g$  は論理式は異なるが, すべての  $A$ ,  $B$ ,  $C$  に対して  $f(A, B, C) = g(A, B, C)$  であり, 同じ論理関数を定義している。

論理関数  $f$  は, 三つの論理変数に対して一つの真理値を決めるので, その定義域と値域を指定するために,  $f: \{0, 1\}^3 \rightarrow \{0, 1\}$  と書かれる。複数の真理値を返す論理関数も定義することができる。一般に,  $n$  個の論理変数に対して,  $m$  個の真理値を返す関数は,  $h: \{0, 1\}^n \rightarrow \{0, 1\}^m$  と書かれる。論理関数  $h$  は, 各次元が真理値をとる  $n$  次元のベクトルに対して, 真理値の  $m$  次元のベクトルを返す関数とみなすことができる。つぎの論理式は, 論理関数  $h: \{0, 1\}^2 \rightarrow \{0, 1\}^2$  を定義する論理式の例である。

$$h(A, B) = (A \cdot \bar{B} + \bar{A} \cdot B, A \cdot B) \quad (1.6)$$

表 1.3 は, 論理関数  $h$  の真理値表である。

表 1.3 論理関数  $h$  の真理値表

$A$	$B$	$h(A, B)$
0	0	(0,0)
0	1	(1,0)
1	0	(1,0)
1	1	(0,1)

### 1.1.1 論理演算の基本法則

論理演算のさまざまな基本法則を, 含まれる論理変数の個数が1個, 2個, 3個の場合に分けて, それぞれ説明する。

〔1〕 1個の論理変数を含む論理演算の基本法則 1個の論理変数  $A$  を含む論理否定, 論理和, 論理積に関する基本法則を示す。これらの基本法則は, 論理演算の意味を考えれば, その正しさは自明である。

単位元:  $A \cdot 1 = A, \quad A + 0 = A$

論理積の単位元は1であり, 1との論理積では値は変わらない。同様に, 論理和の単位元は0であり, 0との論理和では値は変わらない。

零元:  $A \cdot 0 = 0, \quad A + 1 = 1$

論理積の零元は0であり, 0との論理積は必ず0となる。同様に, 論理和の零元は1であり, 1との論理和は必ず1となる。

補元:  $A \cdot \bar{A} = 0, \quad A + \bar{A} = 1$

変数とその論理否定の論理積は, その変数の値にかかわらず必ず0となる。同様に変数とその論理否定の論理和は必ず1になる。

べき等:  $A \cdot A = A, \quad A + A = A$

同じ変数の論理積は, その変数自身となる。論理和も同様である。

二重否定:  $\overline{\bar{A}} = A$

否定の否定は打ち消し合う。

表 1.4 の真理値表より,  $A$  が0と1の場合のそれぞれについて, 各法則の左辺と右辺が等しく, これらの基本法則が正しいことが確認できる。

表 1.4 1変数の基本法則の正しさを確認するための真理値表

A	単位元		零元		補元		べき等		二重否定
	$A \cdot 1$	$A + 0$	$A \cdot 0$	$A + 1$	$A \cdot \bar{A}$	$A + \bar{A}$	$A \cdot A$	$A + A$	$\overline{\bar{A}}$
0	0	0	0	1	0	1	0	0	0
1	1	1	0	1	0	1	1	1	1

# 索

# 引

<b>【あ】</b>	禁止入力	31	最大項	18
アキュムレータ	155	<b>【く】</b>	最大動作周波数	128, 170
アドレスデコーダ	149	組み合わせ回路		<b>【し】</b>
アンダーフロー	95	——の大きさ	15	指数部
<b>【い】</b>		クロック	104	システムリセット
異常	101	——の立ち上がり	104	周期回路
イネーブル付きデコーダ		クロックサイクル	144	主加法標準形
回路	60	クワイン・マクラスキー		主 項
イネーブル付きフリップ		法	35	主乘法標準形
フロップ	110	<b>【け】</b>		出力計算回路
インクリメント回路	74	桁上がり	75	循環小数
<b>【え】</b>		桁あふれ	75	乗算回路
エンコーダ回路	68	桁下がり	82	状態計算回路
<b>【お】</b>		結合法則	6	冗長汎 2 入力ゲート
オーバーフロー		ゲートレベルシミュレー		ジョンソンカウンタ回路
75, 80, 84, 87, 95		ション	124	真理値
<b>【か】</b>		減算回路	82	真理値表
回路記号	8	減算器	85	2, 4, 9
回路図	8	<b>【こ】</b>		<b>【す】</b>
カウンタ回路	121	交換法則	6	ステートマシン
加算回路	77, 168	肯定のリテラル	17	スリーステートバッファ
仮数部	95	固定小数点数	92	回路
数取器	158	<b>【さ】</b>		46, 64
カルノー図	25	再帰的設計	51, 61	スリーステートロジック
完全同期式順序回路	134	サイクリックシフト回路	68	スループット
<b>【き】</b>		サイクルタイム	124	スレーブ D 型ラッチ
記憶回路	99	最小アクセスタイム	113	104
基本ゲート回路	7	最小覆い	28	<b>【せ】</b>
吸収法則	6	最小項	18	正 常
		最小サイクルタイム	127	正の数
		最大アクセスタイム	113	正論理
				83, 100
				積 項
				17
				積分回路
				155
				積和形
				22
				セットアップタイム
				112

セットアップタイム違反 113	バッファ回路 128	
セレクト回路 48, 64	バレルシフト回路 65	【へ】
零 元 5	半加算器 76	べき等 5
全加算器 78	汎 2 入力ゲート 14, 164	変 数 3
<b>【た】</b>	<b>【ひ】</b>	<b>【ほ】</b>
タイミングチャート 101	必須項 28	ホールドタイム 112
畳み込み計算回路 167	ビット 45	ホールドタイム違反 113
立ち上がり検出回路 157	ビット拡張 73, 74	補 元 5
タップ数 167	ビット列 45	<b>【ま】</b>
多入力ゲート 10	ビット列表現 18, 25	マスター D 型ラッチ 104
単位元 5	否定のリテラル 17	マルチプレクサ回路 48
単純型完全同期式順序回路 134	非同期出力 156	<b>【み】</b>
<b>【ち】</b>	非同期セット 107	ミーリー型完全同期式順序回路 134
チャタリング 161	非同期セット・非同期リセット付き D 型フリップフロップ 107	<b>【む】</b>
<b>【て】</b>	非同期セット・非同期リセット付き D 型ラッチ 107	ムーア型完全同期式順序回路 134
デクリメント回路 96	非同期読み出し 104, 148	<b>【め】</b>
デコーダ回路 59	非同期リセット 107	メタステーブル状態 112
デュアルポートメモリ 153	微分回路 156	メタステーブル対策 160
<b>【と】</b>	標準形 22	メモリ回路 148
同期書き込み 104, 148	<b>【ふ】</b>	<b>【ゆ】</b>
同期出力 156	負の数 70	有限オートマトン 138
同期読み出し 151	ファンアウト 10	有効汎 2 入力ゲート 15
動作周波数 127	ファンイン 10	<b>【り】</b>
ド・モルガンの法則 6	符号拡張 74	リテラル 17
ドントケア入力 31	符号付き絶対値表現 70	<b>【る】</b>
<b>【に】</b>	符号なし 2 進数 45	ルックアップテーブル 163
二重否定 5	符号反転 70	<b>【れ】</b>
任意入力 31	符号部 95	レイテンシ 170
<b>【は】</b>	不定値 102	レジスタ 109
ハイインピーダンス 46	浮動小数点数 92, 95	レジスタトランスファレベルシミュレーション 122
排他的論理和 11	部分和計算回路 175	
パイプライン化 171	フリップフロップ 109	
パイプラインレジスタ 171	負論理 83, 100	
ハザード 143	分周回路 117	
バ ス 46	分配法則 6	
発振回路 134, 159		

<p><b>【ろ】</b></p> <p>論理演算 2</p> <p>論理回路 7</p> <p>論理関数 3</p> <p>論理式 2</p> <p>論理指数演算 12</p> <p>論理積 2</p> <p>論理値 2</p> <p>論理否定 2</p> <p>論理変数 3</p> <p>論理和 2</p> <p><b>【わ】</b></p> <p>和 項 17</p> <p>和積形 22</p> <p>ワード 148</p> <p>ワンホットステートマシン 146</p> <p><b>【A】</b></p> <p>AND ゲート 7</p> <p><b>【B】</b></p> <p>BCD (binary coded decimal) 96, 152</p>	<p><b>【D】</b></p> <p>D 型フリップフロップ 104</p> <p>D 型ラッチ 102</p> <p>DSP 回路 173</p> <p><b>【J】</b></p> <p>JK 型フリップフロップ 118</p> <p><b>【N】</b></p> <p><math>n</math> ビットカウンタ回路 121</p> <p>NAND ゲート 11</p> <p>NOR ゲート 11</p> <p>NOT ゲート 7</p> <p><b>【O】</b></p> <p>OR ゲート 7</p> <p><b>【R】</b></p> <p>RS 型フリップフロップ 100</p> <p>RTL シミュレーション 122</p> <p><b>【S】</b></p> <p>SR 型フリップフロップ 100</p>	<p><b>【X】</b></p> <p>XNOR ゲート 13</p> <p>XOR ゲート 13</p> <p><b>【数字・ギリシャ文字】</b></p> <p>0 捨 1 入 94</p> <p>1 の補数 71</p> <p>2 入力セレクタ回路 49</p> <p>2 の補数 72</p> <p><math>2^n</math> 進カウンタ回路 121</p> <p><math>2^n</math> 入力エンコーダ回路 68</p> <p><math>2^n</math> 入力サイクリックシフト回路 68</p> <p><math>2^n</math> 入力バレルシフト回路 65</p> <p>2 進化 10 進数表現 96</p> <p>2 進法 45</p> <p>5 進加算回路 132</p> <p>7 セグメントディスプレイ 33</p> <p>7 セグメントデコーダ回路 33</p> <p><math>\epsilon</math> 時間遅延モデル 123</p>
--	--	---



— 著者略歴 —

中野 浩嗣 (なかの こうじ)  
1987年 大阪大学基礎工学部情報工学科卒業  
1989年 大阪大学基礎工学研究科博士前期課程  
修了 (物理系専攻)  
1992年 大阪大学基礎工学研究科博士後期課程  
修了 (物理系専攻)  
博士 (工学)  
1992年 株式会社日立製作所基礎研究所研究員  
1995年 名古屋工業大学講師  
1998年 名古屋工業大学助教授  
2001年 北陸先端科学技術大学院大学助教授  
2003年 広島大学教授  
現在に至る

伊藤 靖朗 (いとう やすあき)  
2001年 名古屋工業大学工学部電気情報工学科  
卒業  
2003年 北陸先端科学技術大学院大学情報科学  
研究科博士課程前期修了 (情報処理学  
専攻)  
2003年 デンソーテクノ株式会社勤務  
2004年 広島大学助手  
2007年 広島大学助教  
2010年 広島大学大学院工学研究科博士課程後  
期修了 (情報工学専攻)  
博士 (工学)  
2013年 広島大学准教授  
現在に至る

デジタル回路設計入門 —FPGA時代の論理回路設計—

Introduction to Digital Circuit Design —Logic Circuit Design for the FPGA Age—

© Koji Nakano, Yasuaki Ito 2021

2021年4月26日 初版第1刷発行



検印省略

著者 中野 浩 嗣  
伊藤 靖 朗  
発行者 株式会社 コロナ社  
代表者 牛来 真也  
印刷所 三美印刷株式会社  
製本所 有限会社 愛千製本所

112-0011 東京都文京区千石 4-46-10

発行所 株式会社 コロナ社  
CORONA PUBLISHING CO., LTD.

Tokyo Japan

振替 00140-8-14844 · 電話 (03) 3941-3131(代)

ホームページ <https://www.coronasha.co.jp>

ISBN 978-4-339-00943-9 C3055 Printed in Japan

(谷口)



＜出版者著作権管理機構 委託出版物＞

本書の無断複製は著作権法上での例外を除き禁じられています。複製される場合は、そのつと事前に、出版者著作権管理機構 (電話 03-5244-5088, FAX 03-5244-5089, e-mail: info@jcopy.or.jp) の許諾を得てください。

本書のコピー、スキャン、デジタル化等の無断複製・転載は著作権法上での例外を除き禁じられています。購入者以外の第三者による本書の電子データ化及び電子書籍化は、いかなる場合も認めていません。落丁・乱丁はお取替えいたします。