

Verilog HDL で学ぶ コンピュータアーキテクチャ

浅川 毅
四柳 浩之 【共著】
土屋 秀和

コロナ社

まえがき

近年の開発現場では、ハードウェア記述言語（HDL）を用いてデジタル回路設計を行うことが一般的になっている。しかし、デジタル回路設計技術を教える工学系大学では、論理回路による基本的設計技術の教授に留まり、それに続く実践的な HDL を用いた設計について、多くは展開できていない現状にある。そうした中であって、著者らは LSI の設計・評価を研究のメインテーマとして取り組んでいる。関連する LSI メーカーとの技術交流を通して、大学での HDL を用いたデジタル回路設計技術教育の必要性を感じ、講義やゼミ等で実践的教育を行っている。

本書は、コンピュータアーキテクチャの理解をテーマとして、HDL によるデジタル回路設計技術を学ぶ教科書として構成した。2 進数や論理回路などの基本的なコンピュータ工学の基礎は他書に譲り、内容を進めている。また、本書による独習も想定し、とりあげたすべての HDL によるコードは、学習を進める上でそのまま利用できるようにコロナ社 Web サイト (<https://www.coronasha.co.jp/np/isbn/9784339029406/>) で公開しており、回路の FPGA への実装方法（第 11 章）についても解説している。各章には理解度を確認するための演習問題を用意したので、活用していただきたい。

本書を実践的なデジタル回路設計の入門書として、学生や社会人の方々に幅広く利用いただき、技術力向上の一助となれば、著者一同、この上ない喜びである。

2023 年 12 月

著者を代表して 浅川 毅

本書、および本書の Web コンテンツでは、製品名に™、® マークは明記していない。

目 次

第1章 コンピュータアーキテクチャ

1.1 コンピュータの基本構成	1
1.1.1 プログラム内蔵方式	1
1.1.2 コンピュータの基本構成	2
1.2 コンピュータの動作原理	3
1.2.1 CPUとメモリ構成	3
1.2.2 命令の実行	4
1.3 汎用コンピュータの動作の流れ	8
演習問題	13

第2章 マイクロプロセッサとメインメモリ

2.1 マイクロプロセッサ	14
2.1.1 マイクロプロセッサの分類	14
2.1.2 マイクロプロセッサの基本構成	17
2.1.3 直接制御方式	17
2.1.4 間接制御方式	19
2.2 マイクロプロセッサの特性の尺度	21
2.3 メモリ	24
2.3.1 記憶階層	24
2.3.2 メモリの基本構成	25
2.3.3 メモリの特性	26
2.3.4 メモリの分類	27
演習問題	31

第3章 コンピュータの表現と実装

3.1 論理と電気的特性	32
3.1.1 ビットとバイト	32
3.1.2 文字の表現	33
3.1.3 数値の表現	33
3.1.4 負数の表現	35
3.1.5 論理電圧	38
3.2 ステートマシンの表現	39
3.2.1 ステートマシン	39
3.2.2 状態割当	40
3.2.3 状態遷移の制御	41
3.2.4 出力関数の違いによるステートマシンの分類	42
3.3 コンピュータの実装法	46
3.4 FPGA の概要	47
演習問題	50

第4章 Verilog HDL による回路設計

4.1 ハードウェア記述言語を用いる設計の概要	51
4.2 Verilog HDL による論理設計	53
4.2.1 Verilog HDL を使用した設計の流れ	53
4.2.2 モジュールの記述	54
4.2.3 数値と信号の表現	57
4.2.4 回路記述方法	60
4.2.5 回路構成例	63
4.3 Verilog HDL によるシミュレーション	67
演習問題	69

第5章 レジスタ, カウンタ要素

5.1	フリップフロップと順序回路	70
5.2	レジスタ	75
5.2.1	レジスタの回路構成	76
5.2.2	コンピュータで使用する各種レジスタ	77
5.2.3	シフトレジスタ	78
5.3	カウンタ	80
5.4	レジスタの Verilog HDL 記述例	82
5.5	シフトレジスタの Verilog HDL 記述例 1	84
5.6	シフトレジスタの Verilog HDL 記述例 2	86
5.7	カウンタの Verilog HDL 記述例 1	89
5.8	カウンタの Verilog HDL 記述例 2	91
	演習問題	93

第6章 演算要素

6.1	演算装置	95
6.1.1	加算器	96
6.1.2	比較器	99
6.1.3	シフト演算器	102
6.1.4	乗算器	104
6.1.5	データバスと ALU の接続	106
6.2	演算装置の Verilog HDL 記述例	106
6.3	加算器の Verilog HDL 記述例	113
6.3.1	半加算器	113
6.3.2	全加算器	115
	演習問題	117

第7章 制御要素

7.1 デコーダとエンコーダ	118
7.2 マルチプレクサとデマルチプレクサ	121
7.3 制御回路の実装方式	123
7.4 デコーダの Verilog HDL 記述例	125
7.5 エンコーダの Verilog HDL 記述例	127
7.6 マルチプレクサの Verilog HDL 記述例	128
7.7 デマルチプレクサの Verilog HDL 記述例	131
演習問題	132

第8章 コンピュータの命令

8.1 命令と機械語	134
8.2 命令サイクル	135
8.3 命令の形式	136
8.4 アドレスの指定	138
8.5 代表的な命令	141
8.5.1 データ転送命令	141
8.5.2 演算命令	143
8.5.3 比較命令	144
8.5.4 シフト命令	145
8.5.5 プログラム制御命令	145
演習問題	149

第9章 コンピュータの高速化技術と信頼性

9.1 マルチプログラミング	150
9.2 キャッシュメモリ	151
9.2.1 キャッシュメモリの役割	151
9.2.2 キャッシュメモリの構成	152

9.2.3	キャッシュデータの更新	154
9.3	マルチバンク	156
9.4	パイプライン	158
9.4.1	パイプライン処理の原理	158
9.4.2	パイプラインハザード	159
9.4.3	スーパーパイプライン	161
9.5	並列処理	162
9.5.1	スーパースカラ	162
9.5.2	VLIW	163
9.5.3	マルチプロセッサ	164
9.5.4	マルチコアプロセッサ	165
9.6	コンピュータシステムの信頼性	165
9.6.1	RAS	165
9.6.2	MTBF	165
9.6.3	MTTR	166
9.6.4	稼働率	166
9.6.5	システムの稼働率	167
	演習問題	168

第10章 FPGAによるメモリのアクセス

10.1	メモリの構成	170
10.2	メモリコントローラの記述例	177
10.3	メモリ機能チェックシステムの構築	180
	演習問題	186

第11章 FPGAへの実装 (コロナ社Webサイトにて公開)

	付録：設計に用いる命令セット一覧	187
	引用・参考文献	188
	索引	189

コンピュータアーキテクチャ

1940年代に開発されたコンピュータは、社会の要求に応じて、その性能や機能の向上とともに、新たな利用分野を開拓しつつ発展を続けている。そして現在では、家電製品に組み込まれる小型なものからスーパーコンピュータなどの高度なものまで、さまざまなタイプのコンピュータが社会生活の中で利用されている。本章では、これらの基本となるプログラム内蔵方式コンピュータの構成と動作原理について解説する。



1.1 コンピュータの基本構成



1.1.1 プログラム内蔵方式

現在使われているほとんどのコンピュータは、**プログラム内蔵**（stored program）**方式**と呼ばれるコンピュータである。プログラム内蔵方式の概念は、アメリカの数学者ノイマン（J. Von Neumann）により、1945年に「First Draft of a Report on the EDVAC」として発表された。そのためプログラム内蔵方式のコンピュータはノイマン型コンピュータとも呼ばれている。

以下にプログラム内蔵方式コンピュータのおもな特徴を示す。

- ① プログラムやデータは、記憶装置に格納され、アドレスによって指定されてアクセスされる。
- ② 記憶装置に格納された命令は、**プログラムカウンタ**（**PC**：program counter）によって逐次的に指定され、実行がなされる。プログラムカウンタはプログラムレジスタと呼ばれることもある。

1.1.2 コンピュータの基本構成

図 1.1 にコンピュータの基本構成を示す。

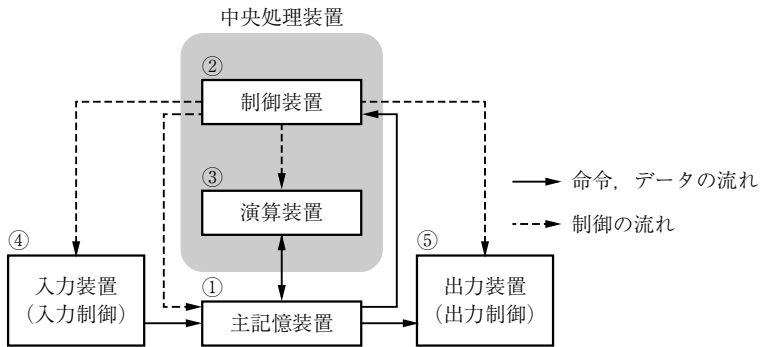


図 1.1 コンピュータの基本構成

① 主記憶装置 (main memory unit)

プログラムやデータを格納する。ハードディスクや CD 等の外部記憶装置 (入出力装置) に保存されたプログラムは、必要に応じて主記憶装置に転送される。実行時には、主記憶装置に格納されている命令が呼び出される。

② 制御装置 (control unit)

主記憶装置から呼び出した命令を解釈して、各装置を制御する。

③ 演算装置 (arithmetic unit)

解釈された命令に従って、算術演算や論理演算を行う。制御装置と演算装置をあわせて**中央処理装置 (CPU: central processing unit)**と呼ぶ。

④ 入力装置 (input unit)

プログラムやデータの入力を行う。

⑤ 出力装置 (output unit)

演算結果などのデータを出力する。



1.2 コンピュータの動作原理



1.2.1 CPUとメモリ構成

図 1.2 に CPU とメモリの基本概念図を示す。

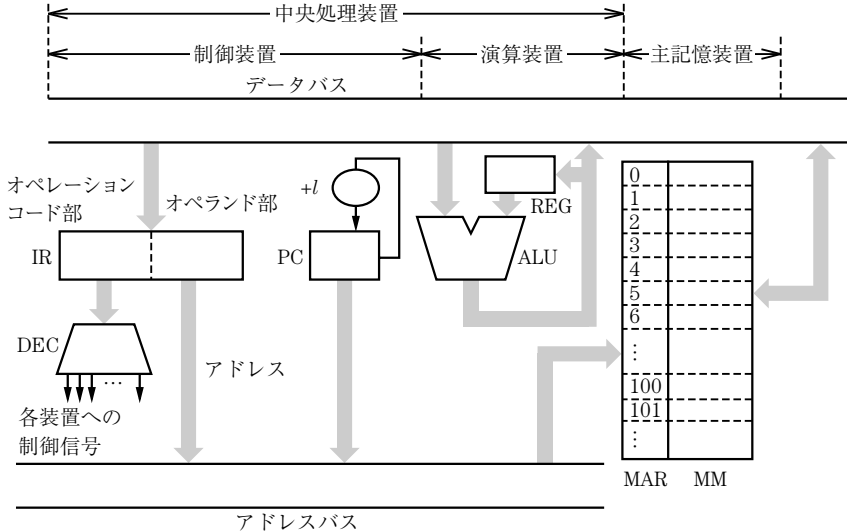


図 1.2 CPU とメモリの基本概念図

① **メインメモリ** (**MM** : main memory)

アドレスで指定された記憶位置に対して、命令やデータの書き込みや読み出しを行う。

② **メモリアドレスレジスタ** (**MAR** : memory address register)

メインメモリへアクセスするためのアドレスを一時的に格納する。

③ **命令レジスタ** (**IR** : instruction register)

メインメモリから読み出した命令を一時的に格納する。処理の内容を示すオペレーションコード部と処理の対象を示すオペランド部によって構成される。

④ **デコーダ** (**DEC** : decoder)

デコーダとは符号化されたデータ（信号）を復号化して元に戻す回路

である。**命令デコーダ** (instruction decoder) は命令を解読し、各装置への制御信号の発生に加えて、プログラムやデータにアクセスするためのアドレスを生成する。

⑤ **プログラムカウンタ (PC : program counter)**

メインメモリに格納されている命令のアドレスを生成する。逐次実行を実現するため、メインメモリから命令が読み出された後、自動的につぎに実行すべき命令が格納されているアドレスを発生する。図 1.2 に示す “+1” は、つぎに実行すべき命令までのアドレスの長さを加えることを意味している。

⑥ **算術論理演算装置 (ALU : arithmetic-logic unit)**

解読された命令に従って、四則演算、比較演算、論理演算などの演算を行う。

⑦ **レジスタ (REG : register)**

演算などに必要なデータを一時的に格納する。

⑧ **データバス, アドレスバス (data bus, address bus)**

図 1.2 において、データバスには、MM, REG, ALU, IR が接続され、アドレスバスには、MAR, PC, IR のオペランド部が接続されている。このように複数の要素 (回路) が共通に扱う信号線の束をバス (バスライン) という。

1.2.2 命令の実行

表 1.1 にプログラム例を示す。これは、メインメモリ (MM) の 100 番地より読み出したデータをインクリメント (+1) して、101 番地に格納するものである。命令 1、命令 3 は 3 バイトで構成され、命令 2 は 1 バイトで構成されるものとする。ここで、命令を構成するために必要なメモリ容量 (バイト数) を、各命令の命令長と呼ぶ。以下、図 1.3 ~ 1.8 を用いて、命令の実行の流れを説明する。

表 1.1 プログラム例

命令番号	MM 格納番地	命令長 (バイト)	命令	処理内容
1	0 ~ 2	3	LD REG, (100)	MM の 100 番地のデータをレジスタ (REG) に転送せよ
2	3	1	INC REG	REG のデータを +1 せよ
3	4 ~ 6	3	ST REG, (101)	REG のデータを MM の 101 番地に転送せよ

〔1〕 命令 1 の読み込み (図 1.3)

プログラムカウンタ (PC) の初期値を 0 とする。このとき、MM の 0 番地、1 番地、2 番地の内容、すなわち 3 バイト分の命令 1 が、命令レジスタ (IR) に読み込まれる。それと同時に PC はつぎの命令の読み込みに備えて 3 加算 (+ 命令長) される。

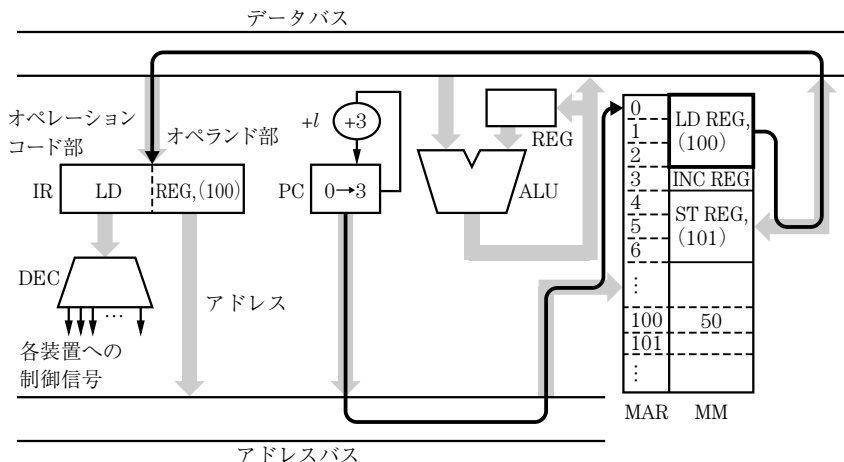


図 1.3 命令 1 の読み込み

〔2〕 命令 1 の解釈、実行 (図 1.4)

IR に格納された命令 1 がデコーダ (DEC) で解釈され、各装置に制御信号を送ると同時に、命令 1 のアドレス 100 がメモリアドレスレジスタ (MAR) へ送られる。この結果、命令 1 「MM の 100 番地のデータ (例: 50) を REG

索 引

【あ】

アイドルタイム 151
 アウト・オブ・オーダー方式
 163
 アクセス 17
 アクセスタイム 26
 アドレスバス 4,17

【い】

インデックス修飾 140

【う】

ウェイ 153

【え】

演算装置 2

【お】

オペレーティングシステム
 10

【か】

間接アドレス指定 139

【き】

キャッシュヒット 154
 キャッシュミス 154
 キャッシュメモリ 151
 競合ミス 155

【け】

桁上げ先見加算器 98
 桁上げ伝搬加算器 98
 桁上げ保存加算器 98

【こ】

構造的ハザード 160

コヒーレンシ 155
 コンパルソリミス 155

【さ】

サイクル 21
 サイクルタイム 26
 算術論理演算装置 4,95

【し】

実効アドレス 141
 指標アドレス指定 140
 指標レジスタ 140
 集中共有メモリ 164
 主記憶装置 2
 出力 17
 出力装置 2
 シリアルアクセス 27

【す】

スカラ 162
 スケジュールング 163
 ステート 21
 スーパースカラ 162
 スーパーパイプライン 161

【せ】

制御装置 2
 制御メモリ 123
 セット 153
 セットアソシエイティブ方式
 152
 セレクタチャネル 20
 全加算器 96
 センスアンプ 29

【そ】

相対アドレス指定 141
 即値アドレス指定 139

【た】

大小比較器 100

【ち】

チップセット 11
 チャネル 19
 中央処理装置 2
 直接アドレス指定 139

【て】

デコーダ 3
 データハザード 160
 データバス 4,17
 電源ユニット 9

【に】

入出力制御装置 19
 入力 17
 入力装置 2

【は】

バイト 32
 バイプライン 159
 バイプラインハザード 159
 パーストモード 20
 ハードウェア記述言語 51
 ハードディスクドライブ 10
 半加算器 96
 バンク 156
 汎用レジスタ 77

【ひ】

ビット 32

【ふ】

ファームウェア 125
 不揮発性 30

フラグレジスタ	77	待ち時間	151	メモリインタリーブ	158
フラッシュメモリ	31	マッピング	152	メモリマップト I/O	17
フルアソシエイティブ方式	152	マルチコアプロセッサ	165	【よ】	
プログラムカウンタ	1,4	マルチバンク	156	容量ミス	155
プログラムカウンタ相対		マルチプレクサチャネル	20	【ら】	
アドレス指定方式	141	マルチプレクスモード	20	ライトスルー方式	155
プログラム内蔵方式	1	マルチプログラミング方式	150	ライトバック方式	155
プロセッサユニット	164	マルチプロセッサ	164	ランダムアクセス	27
分散共有メモリ	164	【み】		【り】	
【へ】		ミラー型	42	リフレッシュ	29
ベースアドレス	140	【む】		リロケータブル	140
ベースアドレス指定	140	ムーア型	42	【れ】	
ベンチマークテスト	22	【め】		レジスタ	4
【ま】		命令アドレス	141	【わ】	
マイクロプロセッサ	14	命令デコーダ	4	ワード	134
マイクロプロセッサ		命令レジスタ	3,77	ワンチップマイコン	16
ユニット	14	メインメモリ	3,11		
マザーボード	9	メモリアドレスレジスタ	3		
マスク ROM	30				

【A】		【D】		【G】	
ADDA 命令	143	DEC	3	GPU	16
ADDL 命令	143	DIMM	11	GR	78
ALU	4,95	DMA	20	【H】	
ASIC	16	DRAM	28	HA	96
【B】		DSA	16	HDL	51
BCD	33,119	DSP	16	【I】	
BIOS ROM	11	DVD ドライブ	10	I/O	17
【C】		DVFS	23	IOP	19
CALL 命令	148	【E】		I/O マップト I/O	18
CISC	14	EEPROM	31	IR	3,77
CLA	98	EPROM	31	【J】	
CLB	47	【F】		JMI	146
CM	123	FA	96	JNZ	146
CPA 命令	144	FIFO	28	JOV	146
CPI	22	FLOPS	22	JPL	146
CPL 命令	144	FPGA	47	JUMP	146
CPU	2,10	FR	77	JZE	146
CSA	98				

	【L】	OS	10	SLL 命令	145
LAD 命令	142			SoC	46
LD 命令	141	【P】		SRAM	30
LIFO	28	PC	1, 4	SRA 命令	145
LSB	33	POP 命令	147	SRL 命令	145
LUT	47	PROM	30	SSD	10, 31
		PUSH 命令	147	ST 命令	142
	【M】			SUBA 命令	144
MAR	3	【R】		SUBL 命令	144
MIMD	164	RAM	28	【V】	
MIPS	22	RAW	160	VLIW	163
MM	3	RCA	98	【W】	
MPU	14	REG	4	WAR	160
MSB	33	RET 命令	148	WAW	160
MTBF	165	RISC	14	【Z】	
MTTR	166	ROM	28	ZF	77
		ROM 書き込み装置	30	【数字】	
	【N】	RTL 記述	51, 64	2 進化 10 進コード	33
NOP	149	【S】			
		SF	77		
	【O】	SISD	158		
OF	77	SLA 命令	145		

— 著者略歴 —

浅川 毅 (あさかわ たけし)
2001年 東京都立大学大学院工学研究科
博士課程修了 (電気工学専攻),
博士 (工学)
2003年 東海大学助教授
2007年 東海大学准教授
2013年 東海大学教授
現在に至る

四柳 浩之 (よつやなぎ ひろゆき)
1998年 大阪大学大学院工学研究科博士
後期課程修了 (応用物理学専攻),
博士 (工学)
1998年 徳島大学助手
2003年 徳島大学講師
2005年 徳島大学助教授
2007年 徳島大学准教授
現在に至る

土屋 秀和 (つちや ひでかず)
2010年 東海大学連合大学院理工学研究科電
気・電子コース博士課程修了 (総合
理工学専攻), 博士 (工学)
2014年 東海大学非常勤講師
2017年 東海大学助教
2022年 東海大学講師
現在に至る

Verilog HDL で学ぶコンピュータアーキテクチャ

Computer Architecture with Verilog HDL

© Asakawa, Yotsuyanagi, Tsuchiya 2024

2024年 2月 20日 初版第1刷発行



検印省略

著者	浅川 毅
	四柳 浩之
	土屋 秀和
発行者	株式会社 コロナ社
	代表者 牛来真也
印刷所	壮光舎印刷株式会社
製本所	株式会社 グリーン

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 978-4-339-02940-6 C3055 Printed in Japan

(西村)



ICOPY < 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、そのつど事前に、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。