

Edition outside of the Philippines is
UNAUTHORIZED AND STRICTLY
PROHIBITED.

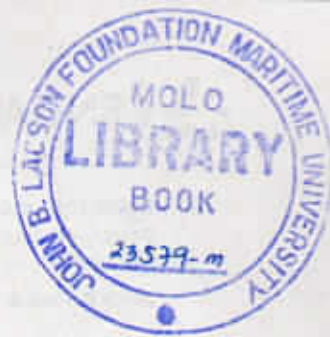
Digital Logic and Microprocessor Design with Interfacing

Second Edition



Enoch O. Hwang

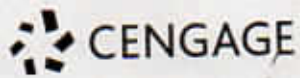
0
005.15
H991
2019



Digital Logic and Microprocessor Design with Interfacing

Second Edition

Enoch O. Hwang



Australia • Brazil • Mexico • Singapore • United Kingdom • United States

CONTENTS

Preface.....	xv
About the Author.....	xxi

CHAPTER 1 Introduction to Microprocessor Design 1

1.1 Overview of Microprocessor Design.....	3
1.2 Design Abstraction Levels.....	6
1.3 Examples of a 2-to-1 Multiplexer.....	7
1.3.1 Behavioral Level.....	7
1.3.2 Gate Level.....	9
1.3.3 Transistor Level.....	11
1.4 Introduction to Hardware Description Language.....	11
1.5 Synthesis.....	15
1.6 Going Forward.....	16
1.7 Problems.....	17

CHAPTER 2 Fundamentals of Digital Circuits 18

2.1 Binary Numbers.....	19
2.1.1 Counting in Binary.....	20
2.1.2 Converting between Binary and Decimal.....	20
2.1.3 Octal and Hexadecimal Notations.....	23
2.1.4 Binary Number Arithmetic.....	25
2.2 Negative Numbers.....	27
2.2.1 Two's Complement Representation.....	27
2.2.2 Sign Extension.....	29
2.2.3 Signed Number Arithmetic.....	30
2.3 Binary Switch.....	32
2.4 Basic Logic Operators and Logic Expressions.....	33
2.5 Logic Gates.....	35
2.6 Truth Tables.....	36
2.7 Boolean Algebra and Boolean Equations.....	38
2.7.1 Boolean Algebra.....	38
2.7.2 Duality Principle.....	41
2.7.3 Boolean Functions and Their Inverses.....	41

2.8	Minterms and Maxterms	46
2.8.1	Minterms	46
2.8.2	Maxterms	49
2.9	Canonical, Standard, and Non-Standard Forms	52
2.10	Digital Circuits	53
2.11	Designing a Car Security System	54
2.12	Verilog and VHDL Code for Digital Circuits	57
2.12.1	Verilog Code for a Boolean Function	57
2.12.2	VHDL Code for a Boolean Function	58
2.13	Problems	59

CHAPTER 3

Combinational Circuits 65

3.1	Analysis of Combinational Circuits	66
3.1.1	Using a Truth Table	67
3.1.2	Using a Boolean Function	70
3.2	Synthesis of Combinational Circuits	72
3.2.1	Using Only NAND Gates	75
3.3	Minimization of Combinational Circuits	76
3.3.1	Boolean Algebra	77
3.3.2	Karnaugh Maps	78
3.3.3	Don't-Cares	85
3.3.4	Tabulation Method	86
3.4	Timing Hazards and Glitches	89
3.4.1	Using Glitches	91
3.5	BCD to 7-Segment Decoder	92
3.6	Verilog and VHDL Code for Combinational Circuits	95
3.6.1	Structural Verilog Code	95
3.6.2	Structural VHDL Code	97
3.6.3	Dataflow Verilog Code	101
3.6.4	Dataflow VHDL Code	102
3.6.5	Behavioral Verilog Code	103
3.6.6	Behavioral VHDL Code	104
3.7	Problems	106

CHAPTER 4

Standard Combinational Components 112

4.1	Signal Naming Conventions	113
4.2	Multiplexer	114
4.3	Adder	117
4.3.1	Full Adder	117
4.3.2	Ripple-Carry Adder	118
4.3.3	Carry-Lookahead Adder	120

4.4	Subtractor	123
4.5	Adder-Subtractor Combination	125
4.6	Arithmetic Logic Unit	129
4.7	Decoder	137
4.8	Tri-State Buffer	140
4.9	Comparator	142
4.10	Shifter	146
4.11	Multiplier	149
4.12	Problems	151

CHAPTER 5

Sequential Circuits 157

5.1	Bistable Element	159
5.2	SR Latch	160
5.3	Car Security System—Version 2	163
5.4	SR Latch with Enable	164
5.5	D Latch	164
5.6	D Latch with Enable	166
5.7	Verilog and VHDL Code for Memory Elements	166
5.7.1	VHDL Code for a D Latch with Enable	168
5.7.2	Verilog Code for a D Latch with Enable	169
5.8	Clock	169
5.9	D Flip-Flop	171
5.9.1	Alternative Smaller Circuit	175
5.10	D Flip-Flop with Enable	176
5.10.1	Asynchronous Inputs	177
5.11	Description of a Flip-Flop	180
5.11.1	Characteristic Table	180
5.11.2	Characteristic Equation	180
5.11.3	State Diagram	180
5.12	Register	181
5.13	Register File	182
5.14	Memories	188
5.14.1	ROM	190
5.14.2	RAM	192
5.15	Shift Registers	197
5.15.1	Serial-to-Parallel Shift Register	199
5.15.2	Serial-to-Parallel and Parallel-to-Serial Shift Register	200
5.15.3	Linear Feedback Shift Register	202
5.16	Counters	205
5.16.1	Binary Up Counter	205
5.16.2	Binary Up Counter with Parallel Load	207

5.17	Timing Issues	210
5.18	Problems	211

CHAPTER 6

Finite-State Machines 215

6.1	Finite-State Machine Models	217
6.2	State Diagrams	221
6.3	Analysis of Finite-State Machines	224
6.3.1	Next-State Equations	225
6.3.2	Next-State Table	226
6.3.3	Output Equations	228
6.3.4	Output Table	228
6.3.5	State Diagram	229
6.3.6	Example	230
6.4	Synthesis of Finite-State Machines	234
6.4.1	State Diagram	235
6.4.2	Next-State Table	236
6.4.3	Next-State Equations	237
6.4.4	Output Table and Output Equations	237
6.4.5	FSM Circuit	238
6.5	Optimizations for FSMs	239
6.5.1	State Reduction	239
6.5.2	State Encoding	240
6.5.3	Unused States	243
6.6	FSM Construction Examples	243
6.6.1	Car Security System—Version 3	243
6.6.2	Modulo-6 Up-Counter	245
6.6.3	One-Shot Circuit	249
6.6.4	Simple Microprocessor Control Unit	251
6.6.5	Elevator Controller Using a Moore FSM	254
6.6.6	Elevator Controller Using a Mealy FSM	258
6.7	Verilog and VHDL Code for FSM Circuits	261
6.7.1	Behavioral Verilog Code for a Moore FSM	261
6.7.2	Behavioral Verilog Code for a Mealy FSM	265
6.7.3	Behavioral VHDL Code for a Moore FSM	266
6.7.4	Behavioral VHDL Code for a Mealy FSM	269
6.8	Problems	270

CHAPTER 7

Dedicated Microprocessors 283

7.1	Need for a Datapath	286
7.2	Constructing the Datapath	287
7.2.1	Selecting Registers	293
7.2.2	Selecting Functional Units	294

7.2.3	Data Transfer Methods.....	295
7.2.4	Generating Status Signals.....	297
7.3	Constructing the Control Unit.....	302
7.3.1	Deriving the Control Signals.....	303
7.3.2	Deriving the State Diagram.....	305
7.3.3	Timing Issues.....	312
7.3.4	Deriving the FSM Circuit.....	315
7.4	Constructing the Complete Microprocessor.....	320
7.5	Dedicated Microprocessor Construction Examples.....	323
7.5.1	Greatest Common Divisor.....	323
7.5.2	High-Low Number Guessing Game.....	330
7.5.3	Traffic Light Controller.....	337
7.6	Verilog and VHDL Code for Dedicated Microprocessors.....	341
7.6.1	FSM+D Model.....	342
7.6.2	FSMD Model.....	351
7.6.3	Algorithmic Model.....	354
7.7	Problems.....	356

CHAPTER 8

General-Purpose Microprocessors 363

8.1	Overview of the CPU Design.....	364
8.2	The EC-1 General-Purpose Microprocessor.....	366
8.2.1	Instruction Set.....	366
8.2.2	Datapath.....	367
8.2.3	Control Unit.....	369
8.2.4	Complete Circuit.....	373
8.2.5	Sample Program.....	373
8.2.6	Simulation.....	375
8.2.7	Hardware Implementation.....	375
8.3	The EC-2 General-Purpose Microprocessor.....	376
8.3.1	Instruction Set.....	376
8.3.2	Datapath.....	377
8.3.3	Control Unit.....	379
8.3.4	Complete Circuit.....	384
8.3.5	Sample Program.....	384
8.3.6	Hardware Implementation.....	387
8.4	Extending the EC-2 Instruction Set.....	388
8.5	Using and Interfacing the EC-2.....	391
8.6	Pipelining.....	395
8.6.1	Basic Pipelined Processor.....	395
8.6.2	Pipeline Hazards.....	397
8.7	Verilog and VHDL Code for General-Purpose Microprocessors.....	399
8.7.1	FSM+D Model.....	399
8.7.2	FSMD Model.....	405
8.8	Problems.....	411

CHAPTER 9

Interfacing Microprocessors 415

9.1	Multiplexing 7-Segment LED Display	416
9.1.1	Theory of Operation	416
9.1.2	Controller Design	417
9.2	Issues with Interfacing Switches	420
9.3	3 × 4 Keypad Controller	427
9.3.1	Theory of Operation	427
9.3.2	Controller Design	429
9.4	PS2 Keyboard and Mouse	431
9.4.1	Theory of Operation—PS2 Keyboard	431
9.4.2	Controller Design—PS2 Keyboard	432
9.4.3	Theory of Operation—PS2 Mouse	436
9.4.4	Controller Design—PS2 Mouse	438
9.5	RS-232 Controller for Bluetooth Communication	444
9.5.1	Theory of Operation—RS-232	445
9.5.2	Controller Design—RS-232	446
9.5.3	Implementation	449
9.6	Liquid-Crystal Display Controller	450
9.6.1	Theory of Operation	450
9.6.2	Controller Design	452
9.7	VGA Monitor Controller	457
9.7.1	Theory of Operation	457
9.7.2	Controller Design	460
9.7.3	Implementation	466
9.8	A/D Controller for Temperature Sensor	467
9.8.1	Theory of Operation	467
9.8.2	Controller Design	469
9.8.3	Implementation	474
9.9	I²C Bus Controller for Real-Time Clock	475
9.9.1	Theory of Operation	475
9.9.2	Controller Design	478
9.9.3	Implementation	483
9.10	Problems	484

APPENDIX A

Xilinx Development Tutorial 486

A.1	Starting ISE	486
A.1.1	Creating a New Project	486
A.1.2	Specifying the FPGA	489

A.2	Creating a New Schematic Source File	492
A.2.1	Drawing Your Schematic Circuit.....	493
A.2.2	Creating and Using a Schematic Symbol.....	495
A.2.3	Editing a Schematic Symbol.....	497
A.2.4	Using a Schematic Symbol in Another Project.....	500
A.3	Creating a New Verilog or VHDL Source File	500
A.4	Setting the Top-Level Module Design File	500
A.5	Mapping the I/O Signals	500
A.5.1	Using PlanAhead for Mapping the Pins.....	501
A.5.2	Faster Alternative Method for Mapping the Pins.....	503
A.6	Synthesis and Implementation	503
A.7	Programming the Circuit to the FPGA	505
A.8	Problems	509

APPENDIX B

Altera Development Tutorial 512

B.1	Starting Quartus	512
B.1.1	Creating a New Project.....	512
B.1.2	Specifying the FPGA.....	514
B.2	Using the Graphic Editor	516
B.2.1	Starting the Graphic Editor.....	516
B.2.2	Drawing Tools.....	516
B.2.3	Inserting Logic Symbols.....	517
B.2.4	Selecting, Moving, Copying, and Deleting Logic Symbols.....	518
B.2.5	Making and Naming Connections.....	519
B.2.6	Selecting, Moving, and Deleting Connection Lines.....	521
B.3	Managing Files in a Project	521
B.3.1	Design Files in a Project.....	522
B.3.2	Creating a New Verilog or VHDL Source File.....	522
B.3.3	Opening a Design File.....	522
B.3.4	Adding Design Files to a Project.....	523
B.3.5	Deleting Design Files from a Project.....	523
B.3.6	Setting the Top-Level Entity Design File.....	523
B.3.7	Saving the Project.....	523
B.4	Analysis and Synthesis	523
B.5	Creating and Using a Logic Symbol	524
B.6	Mapping the I/O Signals	525
B.6.1	Faster Alternative Method for Mapping the Pins.....	527
B.7	Fitting the Netlist and Pins to the FPGA	527
B.8	Programming the Circuit to the FPGA	528
B.9	Problems	529

APPENDIX C

Verilog Summary 533

C.1 Basic Language Elements	533
C.1.1 Keywords	533
C.1.2 Comments	534
C.1.3 Identifiers	534
C.1.4 Signals	534
C.1.5 Numbers and Strings	534
C.1.6 Constants	535
C.1.7 Data Types	536
C.1.8 Data Operators	536
C.1.9 Module	537
C.1.10 Module Parameter	539
C.2 Behavioral Model	540
C.2.1 Assignment	540
C.2.2 initial	541
C.2.3 always	542
C.2.4 Event Control	542
C.2.5 begin-end	544
C.2.6 if-then-else	545
C.2.7 case, casex, casez	545
C.2.8 for	546
C.2.9 while	547
C.2.10 function	547
C.2.11 Behavioral Model Example	548
C.3 Dataflow Model	548
C.3.1 Continuous Assignment	549
C.3.2 Conditional Assignment	549
C.3.3 Dataflow Model Example	550
C.4 Structural Model	550
C.4.1 Built-in Gates	550
C.4.2 User-Defined Module	551
C.4.3 Structural Model Example	552

APPENDIX D

VHDL Summary 553

D.1 Basic Language Elements	553
D.1.1 Keywords	553
D.1.2 Comments	554
D.1.3 Identifiers	554
D.1.4 Data Objects	554
D.1.5 Data Types	554
D.1.6 Data Operators	557
D.1.7 ENTITY	558

D.1.8 ARCHITECTURE.....	559
D.1.9 GENERIC.....	560
D.1.10 PACKAGE.....	562
D.2 Behavioral Model—Sequential Statements.....	564
D.2.1 PROCESS.....	564
D.2.2 Sequential Signal Assignment.....	564
D.2.3 Variable Assignment.....	565
D.2.4 WAIT.....	565
D.2.5 IF-THEN-ELSE.....	565
D.2.6 CASE.....	566
D.2.7 NULL.....	566
D.2.8 FOR.....	567
D.2.9 WHILE.....	567
D.2.10 LOOP.....	567
D.2.11 EXIT.....	567
D.2.12 NEXT.....	567
D.2.13 FUNCTION.....	568
D.2.14 PROCEDURE.....	569
D.2.15 Behavioral Model Example.....	570
D.3 Dataflow Model—Concurrent Statements.....	570
D.3.1 Concurrent Signal Assignment.....	570
D.3.2 Conditional Signal Assignment.....	571
D.3.3 Selected Signal Assignment.....	571
D.3.4 Dataflow Model Example.....	572
D.4 Structural Model—Concurrent Statements.....	573
D.4.1 COMPONENT Declaration.....	573
D.4.2 PORT MAP.....	574
D.4.3 OPEN.....	574
D.4.4 GENERATE.....	575
D.4.5 Structural Model Example.....	575
D.5 Conversion Routines.....	576
D.5.1 CONV_INTEGER().....	576
D.5.2 CONV_STD_LOGIC_VECTOR(,).....	577
Index.....	578

INDEX

\neg , 34
 \vee , 34
+, 34
, 34
 \oplus , 36
 \odot , 36
 Σ , 48
II, 50
 \times , 85
0-maxterm, 49
0-minterm, 47
1-maxterm, 49
1-minterm, 47, 74
7-segment decoder, 92
7-segment LED display, 375, 387, 416

A

Abstraction level, 6
Accumulator, 367
Active edge, 170
Active-high, 113
Active-low, 113
Adder, 117
 carry-lookahead adder, 120
 full adder, 117
 half adder, 205
 ripple-carry adder, 118
Adder-subtractor, 125
Algebra, 38
Algorithmic model, 354
Altera development, 512, *see* Quartus II software
Analog to digital (A/D) converter, 467
Analysis
 combinational circuits, 66
 finite-state machine (FSM), 224
AND, 6, 34, 35, 37
And-of-ors, 43
Application-specific integrated circuits (ASICs), 2, 284
 see also Dedicated microprocessors
Arithmetic logic unit (ALU), 129
Assert, 114
Asynchronous inputs, 177
Ative-high, 113
Axioms, 38

B

Barrel shifter, 156
Behavioral level, 6, 7
 see also Design abstraction levels
Behavioral model, 286, 540, 548, 564, 569
Binary coded decimal (BCD), 92
Binary digit (BIT), 19
Binary numbers, 19
Binary switch, 32
Binary up counter, 205
 with parallel load, 207
Bistable element, 159
Bluetooth, 444
Boolean, 38
 algebra, 38, 77
 function, 41, 70, 72
 theorems, 38
Bus, 4, 285, 296

C

Canonical form, 52
Carry-lookahead adder, 120
Car security system
 version 1, 54
 version 2, 163
 version 3, 243
Central processing unit (CPU)
 see General-purpose microprocessors
Characteristic equation, 180
Characteristic table, 180
Circuit diagram, 9, 53
Clock, 169
 divider, 171
Combinational circuit, 4, 65
 analysis, 66
 minimization, 76
 synthesis, 72
 Verilog code
 behavioral, 103
 dataflow, 101
 structural, 95
 VHDL code
 behavioral, 104

- dataflow, 102
 - structural, 97
 - Combinational components, 113
 - Comparator, 4, 72, 142
 - Control input, 4, 285
 - Control signals, 285, 303
 - Control unit, 3, 285, 302, 364
 - control signals, 285, 303
 - control word, 302, 304, 305
 - FSM Circuit, 315
 - state diagram, 305, 310
 - status signals, 285, 297
 - timing issue, 312
 - see also* Finite-state machine
 - Controller, 2, 286
 - Control word, 302
 - Counters
 - binary up counter, 205
 - binary up counter with parallel load, 207
 - Critical race, 161
- D**
- Dataflow level, 12, 57
 - see also* Design abstraction levels
 - Dataflow model, 57, 101, 102, 548, 570
 - Datapath, 284, 287
 - control signals, 285
 - construction, 287
 - data transfer methods, 295
 - functional units, 294
 - registers, 293
 - need for a datapath, 286
 - register-transfer level (RTL) design, 287
 - status signals, 297
 - Data transfer methods, 295
 - multiple destinations, 296
 - multiple sources, 295
 - tri-state bus, 296
 - De-assert, 114
 - Decoder, 137
 - Dedicated microprocessor, 2, 283
 - algorithmic model, 354
 - Verilog, 355
 - VHDL, 355
 - control inputs, 285
 - control outputs, 285
 - control signals, 285, 303
 - control unit, 285, 302
 - data inputs, 285
 - data outputs, 285
 - datapath, 284, 286, 287, *see also* Datapath
 - examples
 - greatest common divisor, 323
 - high-low number guessing game, 330
 - traffic light microcontroller, 337
 - FSMD Model, 341, 351
 - Verilog code, 352
 - VHDL code, 353
 - FSM+D Model, 341, 342
 - Verilog code, 344
 - VHDL code, 347
 - status signals, 285, 297
 - DeMorgan's Theorem, 39
 - Demultiplexer, 137
 - Design abstraction levels
 - behavioral level, 6, 7
 - dataflow level, 12
 - gate level, 6, 9
 - register-transfer level, 6
 - structural level, 12
 - transistor level, 6, 11
 - D Flip-Flop, 171
 - smaller circuit, 175
 - edge-triggered, 171
 - with enable, 176
 - Digital circuits, 53
 - D latch, 164
 - with enable, 166, 168
 - Don't-cares, 85
 - Dual, 41
 - Duality principle, 41
- E**
- EC-1, 366
 - complete circuit, 373
 - control unit, 369
 - datapath, 367
 - hardware implementation, 375
 - instruction set, 366
 - memory, 368
 - sample program, 373
 - simulation, 375
 - EC-2, 376
 - complete circuit, 384
 - control unit, 379
 - datapath, 377
 - hardware implementation, 387
 - instruction set, 376, 388
 - memory, 385
 - sample program, 384
 - using and interfacing, 391
 - Edge-triggered D flip-flop, 171
 - Elevator controller, 254, 258, 261
 - EEPROM, 2, 188
 - Essential prime implicant (EPI), 83
 - Excitation equation, 370

Excitation values, 218
Execute, 364, 365, 382

F

FA, 117, *see* Full adder
Falling edge, 170
Feedback loop, 158
Fetch, 365, 369
Field-programmable gate array (FPGA),
3, 15
Altera development, 512, 514, 525, 528
Xilinx development, 486, 489, 500, 505
Finite-state machine (FSM), 216
analysis, 224
examples, 230
next-state equations, 225
next-state table, 226
output equations, 228
output table, 228
state diagram, 229
Mealy, 218, 219
behavioral Verilog code, 265
behavioral VHDL code, 269
elevator controller, 258
Moore, 218, 220
behavioral Verilog code, 261
behavioral VHDL code, 266
elevator controller, 254
next-state logic, 217
optimizations
state encoding, 240
state reduction, 239
unused states, 243
output logic, 216
state diagrams, 221, 229, 235
state memory, 216
synthesis, 234
circuit, 238
next-state equation, 237
next-state table, 236
output equation, 237
output table, 237
state diagram, 235
Flip-flop, 158
FPGA, 3,
see Field-programmable gate array
FSM, 215, *see* Finite-state machine
FSMD model, 341, 351
Verilog code, 352
VHDL code, 353
FSM+D Model, 341, 342
Verilog code, 341
VHDL code, 347
Full adder, 117, 118, 126, 130
Function, *see* Boolean function

G

Gate, 6, 35
AND, 35
inverter, 35
NAND, 35
NOR, 35
NOT, 35
OR, 35
XNOR, 35
XOR, 35
Gated D latch, 166
Gated SR latch, 164
Gate level, 6, 9, *see also* Design abstraction levels
General-purpose microprocessors, 284
EC-1, 366
complete circuit, 373
control unit, 369
datapath, 367
hardware implementation, 375
instruction set, 366
memory, 368
sample program, 373
simulation, 375
EC-2, 376
complete circuit, 384
control unit, 379
datapath, 377
hardware implementation, 387
instruction set, 376, 388
memory, 385
sample program, 384
using and interfacing, 391
FSMD model
behavioral FSMD Verilog code, 406
behavioral FSMD VHDL code, 408
FSM+D Model
behavioral Verilog code, 401
structural Verilog code, 400
instruction cycle, 365
overview, 364
program, 373, 384
program counter, 365, 367, 368, 377
Generating status signal, 297
Glitch, 89, 91
Greatest common divisor (GCD), 323

H

Half adder, 205
Half adder-subtractor, 213
Hardware description languages (HDL), 11, 533, 553
Hazards, 397
Hexadecimal number, 24
Hex number, 24
High impedance, 140

High-low number guessing game, 330
 Hold time, 211

I

FC bus controller, 473
 Implementation table, 383
 Instruction cycle, 365
 Instruction encoding
 EC-1, 366
 EC-2, 376
 EC-2 extension, 388
 Instruction register, 365
 Instruction set
 EC-1, 366
 EC-2, 376
 EC-2 extension, 388
 Intel Pentium 4 CPU, 16
 Inverse, 41
 Inverter, 34
 IR, 365
 ISE Design Suite software
 create new project, 486
 FPGA specification, 489
 new schematic source file, 491
 top-level module design file, 500
 Verilog/VHDL source file, 500
 Iterative circuit, 143

J

JK flip-flop, 213

K

Karnaugh map, 78
 Keypad controller, 427
 Keywords
 Verilog language, 533
 VHDL, 553
 K-map, 78

L

Latch, 158
 Level-sensitive, 169
 Linear feedback shift register, 202
 Liquid-crystal display (LCD), 450
 Logic expression, 33
 Logic gates, 35, *see also* Gate
 Logic operator, 33
 Logic symbol, 7, 35, 495, 524

M

Master-slave D flip-flop, 172
 Maxterm, 46, 49

Mealy FSM, 217, 258
 Memories
 RAM, 192
 ROM, 190
 Microcontrollers, 2, 286
 Microprocessor, 2
 control unit, 285, 302
 datapath, 284, 286, 287
 dedicated, 2, 284
 design overview, 3
 general-purpose, 2, 284, 364
 Minimization of combinational circuits, 76
 Boolean algebra, 77
 K-map, 78
 Quine-McCluskey, 87
 tabulation, 86
 Minimize 1-bit scheme, 240
 Minimum bit change scheme, 240
 Minterm, 46
 Modulo-6 up-counter, 245
 Monitor, 457
 Moore FSM, 217, 254
 Multiplexer, 7, 114, 538
 Multiplier, 149
 MUX *see* Multiplexer

N

NAND gate, 35, 75
 Negative edge-triggered flip-flop, 172
 Negative logic, 113
 Negative numbers, 27
 signed number arithmetic, 30
 sign extension, 29
 two's complement representation, 27
 Netlist, 12, 15
 Networks, 53
 Next-state equations, 225, 237
 Next-state logic circuit, 4, 217, 218, 285
 Next-state table, 226, 236
 NMOS, 139, 141
 Nonstandard form, 53
 NOR gate, 35
 NOT gate, 6, 34, 35, 37

O

Octal numbers, 23
 One-hot encoding scheme, 242
 One-shot circuit, 91, 249
 Opcode, 364
 Optimization of FSM circuits, 239
 state encoding, 240
 state reduction, 239
 unused states, 243
 OR, 6, 34, 35, 37

Or-of-and, 42
 Output equations, 228, 237
 Output logic circuit, 4, 216, 218, 285
 Output table, 228, 237

P

Parallel-to-serial shift register, 200
 Period, clock, 170
 Pipelining

- basic pipelined processor, 395
- hazards, 397

 PMOS, 138, 141
 Positive edge-triggered D flip-flop, 171
 Positive logic, 113
 Prime implicant (PI), 83
 Prioritized adjacency scheme, 240
 Product-of-maxterms, 49
 Product-of-sums, 43
 Product term, 41, 46, 77
 Program counter (PC), 365, 368, 377
 PS2 keyboard, 431
 PS2 mouse, 436

Q

Quartus II software, 512

- analysis and synthesis, 523, 527
- create new project, 512
- FPGA specification, 514
- graphic editor, 516
- logic symbol, 516, 524
- managing files, 521
- pin mapping, 525, 527
- programming the FPGA, 528
- setting top level entity, 523

 Quine-McCluskey method, 76, 87

R

RAM, 192
 Random numbers, 202
 Real-time clock, 475
 Register, 181
 Register file, 182
 Register transfer, 6, 171, 287, 312
 Ripple-carry adder, 118
 Rising edge, 169
 ROM, 190
 Rotator, 147
 RS-232, 444
 RTL, *see* Register transfer

S

Schematic diagram, 9, 35, 53
 Sequential circuits, 66, 157

- bistable element, 159

characteristic equation, 180
 characteristic table, 180
 clock, 169
 Flop-flop, 171
 latch, 160
 memories, 188
 register, 181
 register file, 182
 shift registers, 197
 state, 216
 state diagram, 180
 timing issues, 210
 Serial communication, 431, 444, 475
 Serial-to-parallel shift register, 199, 200
 Setup time, 211
 Shifter, 146
 Shift registers, 197

- linear feedback, 202
- serial-to-parallel and parallel-to-serial, 200
- serial-to-parallel, 199

 Signal naming conventions, 113
 Signed numbers, 27
 Sign extension, 29
 Simulation trace, 56
 SR latch, 160
 Standard combinational components

- adder
 - carry-lookahead adder, 120
 - full adder, 117
 - ripple-carry adder, 118
- adder-subtractor combination, 125
- arithmetic logic unit (ALU), 129
- comparator, 142
- decoder, 137
- multiplexer, 114
- multiplier, 149
- shifter, 146
- subtractor, 123
- tri-state buffer, 140

 Standard form, 53
 State, 158, 216
 State diagram, 221, 229, 235
 State encoding, 240
 State memory, 4, 216, 218
 State reduction, 239
 Status signals, 285, 297
 Structural level, 12

- see also* Design abstraction levels

 Structural model

- Verilog, 550
- VHDL, 573

 Subcube, 80
 Subtractor, 123, 125
 Sum of minterms, 47
 Sum-of-products, 42
 Sum term, 43
 Switch, 32, 420

Synthesis, 15

- Altera development, 523, 527
- combinational circuits, 72
- finite-state machines, 234
- Xilinx development, 503

Synthesizer, 6, 11, 15

T

- Tabulation method, 76, 86
- Temperature sensor, 467
- Theorems, 38
- Timing hazards, 89
- Timing issues, 210, 306, 312
- Timing trace, 57
- Traffic light controller, 337
- Transistor, 6, 11
- Transistor level, 6, 11
- Transmission gate, 11
- Transparent latch, 166
- Tri-state buffer, 140
- Tri-state bus, 296
- Truth table, 36, 67, 72
- Two's complement representation, 27

U

- Unsigned numbers, 19, 27
- Unused states, 243, 246

V

- Verilog, 7, 11
 - algorithmic model, 341, 354

- behavioral, 8, 95, 103, 261, 263, 265
- dataflow, 95, 101
- dedicated microprocessor, 341, 344, 352
- FSM+D, 342, 344, 399
- FSMD, 351, 352, 405
- general-purpose microprocessor, 399
- memory element, 169
- structural level, 95
- syntax summary, 533

VGA monitor controller, 457

VHDL, 7, 11

- algorithmic, 341, 354
- behavioral, 9, 95, 104, 261, 266, 269
- dataflow, 95, 102
- dedicated microprocessor, 341, 347, 353
- FSM+D, 342, 347, 399
- FSMD, 351, 353, 405, 408
- general-purpose microprocessor, 399
- memory element, 168
- structural, 95, 97
- syntax summary, 553

X

- Xilinx development, 486, *see* ISE Design Suite software
- XNOR gate, 35
- XOR gate, 35

Z

- Z, 137, 296, 534, 545, 549, 555