<table>
<thead>
<tr>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-bit down counter, state graph, 137–8</td>
</tr>
<tr>
<td>32-bit down counter, verification, 130–1</td>
</tr>
<tr>
<td>32-bit multiplier, 126–7, 136–7</td>
</tr>
<tr>
<td>abstract bus channels, 56</td>
</tr>
<tr>
<td>Accellera OVL, 99</td>
</tr>
<tr>
<td>adaptive branching heuristics, 255</td>
</tr>
<tr>
<td>adequacy, 128</td>
</tr>
<tr>
<td>advanced microcontroller bus architecture (ARMA), 90</td>
</tr>
<tr>
<td>Algebraic Decision Diagrams (ADDs), 182</td>
</tr>
<tr>
<td>alias reduction, 83</td>
</tr>
<tr>
<td>Alpha microprocessor, 22</td>
</tr>
<tr>
<td>AMBA APB protocol, 102</td>
</tr>
<tr>
<td>AMBA models</td>
</tr>
<tr>
<td>accuracy, 64</td>
</tr>
<tr>
<td>performance, 65</td>
</tr>
<tr>
<td><em>analyze-conflict</em> operator, 7</td>
</tr>
<tr>
<td>and-inverter graphs (AIG), 21, 250, 255</td>
</tr>
<tr>
<td>APPLY algorithms, 174–94, 178</td>
</tr>
<tr>
<td>for <em>BMDs</em>, 185–6</td>
</tr>
<tr>
<td>for BDDs, 199</td>
</tr>
<tr>
<td>see also composition algorithms and rules</td>
</tr>
<tr>
<td>arc-consistency, 250–1</td>
</tr>
<tr>
<td>architectural-level design, 92, 122</td>
</tr>
<tr>
<td>arithmetic/logic unit (alu) verification, 131–3, 142</td>
</tr>
<tr>
<td>assertions, 99–102, 158</td>
</tr>
<tr>
<td>in BUS monitor example, 102–11</td>
</tr>
<tr>
<td>for interrupt controller verification, 169</td>
</tr>
<tr>
<td>and observability, 97</td>
</tr>
<tr>
<td>audit process, 124, 126</td>
</tr>
<tr>
<td>automated theorem proving, 125</td>
</tr>
<tr>
<td>automatic test pattern generation (ATPG), 8–11, 143, 148–9, 257–9</td>
</tr>
<tr>
<td>word-level techniques, 189</td>
</tr>
<tr>
<td>average path length (apl), 231–2, 236–40</td>
</tr>
<tr>
<td><em>backtrack</em> operator, 7</td>
</tr>
<tr>
<td>backtracking, 7</td>
</tr>
<tr>
<td>chronological, 7</td>
</tr>
<tr>
<td>non-chronological, 252, 253</td>
</tr>
<tr>
<td>with clause learning, 251</td>
</tr>
</tbody>
</table>

*see also* conflict-driven clause learning (CDCL); SAT solvers  
backtracking, chronological, 7  
backward model-checking, 17–18  
base objects definition, 163  
BDD bounding, 19–20  
behavioral designs, 204–5  
behaviors, 54  
in component-assembly model, 55–6  
timing of, 56  
benchmarks  
IWLS’93, 233  
MCNC, 233  
binary decision diagrams (BDDs), 3–6, 173, 174, 175–81  
applications, 180  
and Boolean networks, 125  
canonicity, 4  
and combinational equivalence checking, 180  
complemented edges, 180  
construction, 177, 178  
definition, 176  
extensions, 180  
isomorphic, 4, 176  
as linear inequality encoding, 250  
low edges, 180  
and model checking, 17–18  
reduction rules, 4  
size, 4–5, 179, 181  
software for, 181  
variable reordering, 4–5, 224  
and word-level specifications, 182  
binary decision trees, 3  
binary decomposition, 173  
binary moment diagrams (BMDs), 173, 174, 183–8  
binary time-frame expansion, 22  
Boole function expansion, 174  
Boolean circuits, 249–50  
conversion to CNFs, 249  
structural sharing, 250  
Boolean Constraint Propagation (BCP), 7, 248  
Boolean decomposition, 181–2  
Boolean Expression Diagrams (BED), 250  

© Cambridge University Press  
www.cambridge.org
Boolean reasoning techniques, 2–11

Boolean satisfiability (SAT) problem, 6–8, 188
algorithms for, 251–5
using BDDs, 180–1
for CNFs, 248
extensions, 246–7, 256–7
multi-valued, 227
NP-completeness, 246
see also SAT solvers

boundary scan, 117
bounded cone of influence (BCOI) reduction, 21
bounded model checking, 20–2, 260–2
branch coverage (BC) metrics, 133
bridge faults, 143
bus-arbitration TLM model, 53, 56–8, 65
bus-functional TLM model, 53, 58, 65
bus transaction block, 62
buses, 66

C/C++ languages, 35–47, 94
classes, 52
equivalence checking, 36
caches, 113, 114, 120
canonicity, 173
of graphs, 173–5
of ROBDDs, 176
cardinality, 251
cardinality constraints, 250
CCATB models, 90
CF-SMODDs, 230
average path length, 232
evaluation, 231, 240
Chaff SAT solver, 8
characteristic functions, 17, 30–1
evaluation, 231
in GF(N), 229–30
check tables, 128, 140
checkers, 155, 156, 157–8
children, 151
chopping see program slicing
chronological backtracking, 7
circuit activity metrics, 136–7
circuit delay computation, 263
circuit observability don’t cares, 255
circuits, finite field representation, 214
clause deletion policies, 255
clause learning, 251
see also conflict-driven clause learning (CDCL); SAT solvers
closed-loop stimuli generation, 145, 146, 148
co-factor operation, 5
co-NP-hard problem, 26
code coverage metrics, 131–6, 144
co-factors, 216
combinational equivalence checking (CEC), 23, 24–8, 180, 257
see also equivalence checking; sequential equivalence checking
combinational portions, 23, 24–6
complete assignment, 247
complex numbers, and TEDs, 209
component-assembly TLM model, 53, 55–6, 65
composition algorithms and rules, 174–94, 199–202, 227
see also APPLY algorithms
computational tree logic (CTL), 13, 125, 260
computational trees, 12
compute EU procedure, 14
concurrent design behavior, 94–5
condition coverage (CC) metrics, 133–4
cone-of-influence (COI), 21, 95
conflict analysis, 7–8
conflict-driven clause learning (CDCL) SAT solvers, 252–5
conjunctive normal form (CNF), 6, 19, 247
and BDDs, 3
for BMC, 21
conversion from Boolean circuits, 249
and linear equalities, 250–1
preprocessing of, 255–6
satisfiability, 248
connected graphs see state graphs
constrained random stimuli generation, 159
contrapostum, law of, 9–10
circuit design blocks, 95
circuit flow graphs, 134–6
test controllability, 96
cooperating validity checker (CVC), 35, 37–8
corner cases, 127, 147, 159, 160
coverage collector, 99
coverage groups, 157
coverage measurement, 129
coverage metrics, 128–30, 154
choice of, 130, 144
classification of, 130–44
ease of computation, 144
and manufacturing faults, 143
coverage model, 130–1
coverage monitors, 157, 169
coverage-directed stimulus generation, 157
CPU simulators, 118
cross-over operators, 151
cross-talk operators, 143
cross-talk noise, 263
CTL, 12
CT*, 12
CTL’, 12–13
CUDD package, 181, 205
cut points, 190
cycle-accurate computation model, 53, 54, 58–60
cycle-accurate simulators, 114, 118
D-Algorithm (D-ALG), 148–9
dataflow graphs, 175, 212
data forwarding, 150
data transform blocks, 95, 96
data transport blocks, 95–6
datapath design blocks, 95–6
Davio decomposition see XOR decomposition
Davis Punam Logemann–Loveland (DPLL) procedure, 6–8, 252, 253
debuggers, 118–19, 263
decide-next-branch loop, 7
decision assignments, 252
decision coverage see branch coverage (BC) metrics
decision diagrams, 174, 181–2
in GF(N), 218
see also algebraic decision diagrams; binary decision diagrams; finite field decision diagrams; multiple-output decision diagrams; multi-valued decision diagrams; reduced ordered binary decision diagrams; word level decision diagrams
decision levels, 252
decision procedures, 37
decomposition principle, 173–4
for BDDs, 176–8
for BMDs, 183–4
see also Boolean decomposition; first moment decomposition; XOR decomposition
deduce operator, 7
delay faults, 143
design abstraction levels, 122–3
design debugging, 259–60
design engineers, 125, 155, 160
design flow overview, 36
design under verification (DUV) layer, 154
interrupt controller verification, 160–3
testbench component, 97, 99, 155, 156
design verification, definition, 124
deterministic stimuli generation, 148–9
device under test (DUT), 124, 126, 129, 143
directed acyclic graphs (DAG), 175
directed graphs, 15, 175
directed stimuli generation, 146–7, 159
discrete cosine transform, 207, 209
disturbing influence, 64
double handshake, 67–9
drivers, 99, 159, 167
dynamic reordering, 5
easily invertible form, 181
ECF-SMODD, 230, 231, 232
average path length, 232
evaluation, 231–2, 240
edge traversing probability, 231, 232
edge-valued BDDs (EVBDDs), 182
size complexity, 200
elliptic wave filter, 207, 209
embedded systems, debugging, 113–14, 119
overview, 113–14, 120–1
tools, 114–19
functional, 119
performance-oriented, 119
emulation, 96, 115, 117
coded characteristic functions (ECFs), 230
ergy consumption see power consumption
equivalence checking, 2, 125, 190
for C descriptions, 36
high-level, 34–47
and textual differences, 39
word-level, 189–91
see also combinational equivalence checking;
sequential equivalence checking
error-based coverage metrics see fault-based coverage metrics
evaluator module, 146
evolutionary algorithms, 151
explicit-state model checking, 260
exponential, and TEDs, 210
expression coverage (EC) metrics, 134
failing property, 261
c fast Fourier transform, 207, 209
fault-based coverage metrics, 141–3
FDIV bug, 124
feedback-based stimuli generation, 145–6
feedback-driven stimuli generation, 146
feedback-shifted stimuli generation, 146
field-programmable gate arrays (FPGAs), 117
FIFO buffers, 77
filtercore function, 83
finite field decision diagrams (FFDDs), 174, 175
finite fields,
applications, 213
characteristic, 215
definition, 214
generation of, 215
generation of, 215
graph representation, 217–20
notation for, 216
operations on, 216
and representation of circuits, 214
see also Galois fields
finite state machines (FSMs), 23
for microprocessor core verification, 149
reachable states, 29–31
for sequential checking, 29–30
traversal, 180
finite state machine-based metrics, 137–40, 144
FIR filter, 207, 209
first moment decomposition, 202
floating-point operations see multiplicative power
hybrid decision diagrams,
formal verification, 1–2, 96, 124–5
formality tool, arithmetic proof engine (APF), 190
forward model-checking, 17
functional correctness checker, 157–8
functional coverage metrics, 128, 140–1, 144
functional debugging, 119
functional dependencies, 263
functional verification, 92–7, 110, 129
Galois fields, 175, 209, 211, 214
see also finite fields
gate design level, 123
gates, 125, 249
GAUT architect synthesis tool, 204, 205, 207
generations, 151
Genesys test data generator, 150–1
golden model, 143
gprof profiler, 118
granularity, and simulation speed, 52–60, 62–3
handshakes, 67–9, 147
hardware description languages (HDL), 123, 154
DUV descriptions, 155, 156
simulators, 117
hardware verification languages (HVL), 155
high-level design specification, 154
high-level logics, 189
Huffman decoding, 83
i801 microprocessor, 138–40, 141
IEEE 1850 PSL standard, 99–102, 111
IIR filter, 207, 209
image computation, 31–2
image operation, 17, 19–20
IMDCT function, 83
Img operator, 17
implementation model, 53, 60
implication graphs, 8, 252–3
implicit-state enumeration, 180
implicit-state model checking, 260
in-circuit emulators, 115, 117, 119
induction with depth, 20
inductive reasoning, 19, 20
inductive variant, 20
infinite precision computation model, 204
inferring rule, 19
input monitors, 156, 168
instruction pipeline, 113
Intel Vtune profiler, 118
internal equivalences, 26
interrupt controller case study, 160–71
invariants see safety properties
IPSIM framework, 90
irritators, 159, 160
isomorphism, 173, 176
ite operator, 5–6
IWLS’93 benchmark, 233
justification frontiers, 255
K*BMD diagrams, 188, 200
Kalman filters, 207, 209
Kernighan, Brian, 129
keys, 205
Kripke structures, 11–12
for bounded model checking, 260
for CTL model checking, 14, 15, 16
latches, 23–4
correspondence relation, 24
mapping problem, 23
lazy data structures, 254
learning, in ATPG, 9–10
least mean square computation, 207, 209
linear inequalities, 250–1
linear-time temporal logic (LTL), 125, 260
literal-based expansion, 217
liveness, 14, 21
local search, 251
logic analyzers, 115–16, 116–17, 119
logic of counter arithmetic with lambda expressions
and uninterpreted functions (CLU), 189
logic of equality with uninterpreted functions (EUF), 189
logic simulation, 1
see also simulation
manager, 205
manual stimuli generation, 146–7
manufacturing faults, 8–9, 257–9
coverage metrics, 143, 144
Maple, 190
Mathematica, 190
MatLab, 190
matrix product computation, 207, 209
maximum satisfiability (MaxSAT), 246, 256
MCNC benchmark, 233
memories, 66
metrics see coverage metrics
microarchitectural specification, 92
Microblaze processor, 83, 83–5
MIN-MAX post algebra, 214, 217, 218
meters, 25, 26, 257, 258
model checking, 2, 11–22, 125
for BDDs, 17–18
bounded, 260–2
model counting, 246
moment decomposition, 174, 183, 192
monitors, 99, 154, 155, 156–7, 169
Moore machines, 12
MP3 decoder, 83–5
block diagram, 83
design validation, 88
development, 85–8
multi-hop transactions, 79–80
multiple coverage metrics, 129
multiple-input multiple-output functions, 227
multiple-output functions, finite field representation, 212–13
evaluation of, 230–2
see also multiple-output decision diagrams
multiple-output decision diagrams (MDDS),
algebraic operations, 225–7
canonicality, 220–2
composition rules, 227
evaluation time, 236–40
form of, 218
and GF($N$) functions, 218–20
minimality, 222
and multiple-input multiple-output functions, 227
and multi-valued SAT, 227
performance, 233–6
reduction, 220, 222–3, 228–9
variable reordering, 223–5
multi-hop transactions, 79
multiple-valued literals, 217
multiplicative binary moment diagrams (*BMDs),
184–8
and APPLY algorithm, 185–6
and Boolean logic, 186
and Boolean satisfiability problem, 188
reduction rules, 184–5
normalization, 185
size complexity, 200
signal processing applications, 207, 209
and Taylor expansion diagrams, 205
variable ordering, 185
variants, 188
and word-level verification, 187–8
multiplicative diagrams, 184
multiplicative power hybrid decision diagrams
(PHDDs), 188
multiprocessors, debugging, 114
multi-terminal BDD (MTBDD), 182
size complexity of, 200
multi-valued decision diagrams (MDDS), 213, 233
multi-valued decomposition, 174
multi-valued SAT problem (MV-SAT), 227
mutant descriptions, 142
mutation coverage (MC) metrics, 141–2, 144
mutation operators, 151
NiVER resolution package, 255–6
nodes,
nomenclature, 193
potentially equivalent, 27–8
reduction of see reduction
redundant, 194
node traversing probability, 231, 232
non-chronological backtracking, 252, 253
non-linear design blocks, representation, 213
NP-complete problems, 5, 6, 10, 246
observability, 92, 96, 111
observability-based code coverage metric (OCCOM), 143–4
open-loop stimuli generation, 145, 151
optimistic modeling, 64
optimistic prediction, 63
ordered binary decision diagrams (OBDDs), 176, 181–2
reduction of, 177
ordered functional decision diagrams (OFDDs), 181–2
ordered Kronecker FDDs (OKFDDs), 182
oscilloscopes, 115, 116–17
OSI layer implementation, 90
output monitors, 157, 169
parents, 151
partial assignment, 247, 250–1
partial MaxSAT, 256
pattern generators, 116
path coverage (PC) metrics, 134–6
path quantifiers, 12
path-oriented decision making (PODEM), 149
PC sampling, 117
PCI express data link layer block, 95
Pentium processor, 124
performance-oriented debugging, 119
pipelined processors, 263
platforms
constituent objects, 66
debugging of, 113–14, 120
as debugging tool, 115
PLOVER-PODEM algorithm, 149
polling, 69
port interface, for interrupt controller verification, 163
power consumption, 113, 114, 118
power measurement, 116–17
power simulators, 114, 118
PowerPC microprocessor, 22
preprocess operator, 7
preprocessing, 255–6, 259
Presburger arithmetic, 189
primitive elements, 215
processes, 66, 80–3
product terms, 217, 219
profilers, 114, 117–18, 119–20
program slicing, 38–9
properties, 14
property specification language (PSL), 94, 99
Index

propositional formulas, 247
assignments, 247–8
see also decision assignments
protocol channels, 58
protocol checkers, 157
protocol refinement, 58
pseudo-Boolean constraints, 246, 256
PSPACE-completion decision problems, 256
quantified boolean formulas (QBFs), 256–7
quantifier-free logic, 189
random resistant circuits, 150
random stimuli generation, 149–51, 159
reachability analysis, 29
reactive systems, 94
real time-accurate models, 58
recursive expansion, 218–19
recursive learning, 10–11, 251
Red–Miller decomposition see XOR decomposition
reduce function, 5
reduced boolean circuits (RBC), 250
reduced ordered binary decision diagrams (ROBDDs), 176
canonicity, 176
isomorphism, 176
Shannon decomposition, 177–8
reduction of *BMDs, 184–5
of BDDs, 4, 176
of MODDs, 220, 222–3, 228–9
of OBDDs, 177–8
of ROBDDs, 177–8
reference models, 157–8, 158–9
register transfer level (RTL) models, 125
as design level, 122
development time, 88
simulation speed, 51, 127
relational operators, and TEDs, 211
rendezvous synchronization see double handshake resolution, 248–9, 255
resolution refutation, 255
resource contention, 57–8, 65
resource sharing, 190–1
responders, 99
response checking, 99, 126, 129
restriction operation, 5
result oriented modeling (ROM), 63–4, 89
routing, of transactions, 79–80
safety properties, 14, 16–17, 21, 260–1
SAT solvers, 6–8
applications, 246, 263
auxiliary functions, 253–4
bounded model checking, 20–2
Chaff, 8
SATO, 8
symbolic model checking, 18–20
see also Boolean satisfiability problem
satisfiability see Boolean satisfiability problem
satisfiability modulo theories (SMT), 246, 257
satisfying assignment, 6
SATO SAT solver, 8
SBDDs, evaluation time, 236–40
scoreboards, 99, 155, 157, 158–9
sequential depth, 22
sequential design behaviour, 94
sequential equivalence checking (SEC), 23, 28–34
see also combinational equivalence checking; equivalence checking
Shannon expansion, 174, 177–8, 217
shared MODDs (SMODDs), 227–8
average path length, 232
evaluation, 230–2, 240
variable reordering, 228
shared sub-structures, 255
sifting, 5
signal processing applications
for *BMDs, 207, 209
for TEDs, 207–9
similarity of designs
simple induction, 20
SimplePower, 118
SimpleScalar, 118
simulation
of design behavior, 96
flow of, 154
speed of, 130–1, 154–5
simulation-based verification, 126–8
simulators
debugging of, 120
as debugging tool, 114
single stuck-at faults, 143
single tag models, 144
slave devices, 99
slicing, 38–9
SMV verification system, 181
software debuggers, 114
sparse recursive representation, 194
SpecC, 89
special cases, 127, 147
specification models, 53, 54, 88
state coverage metric (FSM-SC), 137
state explosion problem, 16
state graphs, 137–8, 140
state transition graphs, 29
statement coverage (SC) metrics, 128, 131–3
stimuli, 145, 155, 156, 159
stimuli sets, 126, 127–8
generation of, 99, 129, 145–51, 157
structural sharing, Boolean circuits, 250
structural similarity, 26–7
stuck-at faults, 9, 143, 149, 258–9
sum-of-products forms, 3
swap-based algorithm see binary decision diagrams, variable reordering
symbolic abstraction, 192
symbolic algebraic design methods, 189–90
symbolic simulation, 35, 36–7
symbolic trajectory evaluation, 263
synthesis oriented TLMs, 65–6, 88
system LSIs, 1
system on a chip (SoC), 1
SystemVerilog Assertion (SVA) package, 94, 100, 111, 123
interrupt controller case study, 160, 163–71
SystemC, 52, 89–90, 94
processes, 80–3
TLMs for, 80–8
SystemCSV, 89
tags, 144
Tarjan’s algorithm, 15
task graphs, 89
tautology, 181
Taylor expansion diagrams (TEDs), 174–5, 192–212
algorithmic-level verification, 209
for array processing, 206–7
for behavioural HDL, 204–5
and +BMDs, 205, 206
and Boolean logic, 202, 206
canonicity, 196, 197
and complex numbers, 209
composition rules, 199–202
decomposition, 193–4
definition of, 193
and exponentials, 210
functional equivalence verification, 204
and Galois fields, 209
high-level transformation verification, 206
isomorphism, 194
limitations, 209–11
minimality, 197
normalization, 195
reduction, 194–5
and relational operators, 211
and RTL verification, 202–4, 205, 206
signal processing applications, 207–9
size, 197–9, 205, 206
static ordering, 205
uniqueness, 196
variable ordering, 205
Taylor series, 192–3
uniqueness, 196
technology mapping, 263
TEDify software, 205
temporal logics, 12, 260
temporal operators, 12
term rewriting systems, 189
test controllers, 97–9
test criterion, 128
test generation tools, 115
test quality, 126
test sets, 126
testbenches,
analysis communication, 110–11
components, 97, 99, 155–60
and error identification, 96–7
for interrupt controller case study, 163–71
textual differences, 36, 39
threshold stimuli generation, 146
TI OMAP multiprocessor, 114
timing bugs, 120
toggle coverage (TC) metrics, 136–7, 144
takes, 119
transition coverage metric (FSM-TC), 137
transaction generators, 159
for interrupt controller verification, 163–7
transaction interfaces, 159
transaction level modeling (TLM)
accuracy of, 64–5
developments in, 89–90
estimation oriented, 62–5
synthesis oriented, 65–6
taxonomy, 51–61
types of, 51–61
transactions, 159
transactors, 97
transducers, 66, 75–8
transformational systems, 94
transistor design level, 123
transition coverage see finite state machine
transition coverage (FSM-TC)
transition faults, 143
truth tables, 3
Tseitin’s transformation, 249
UML language, 89
unbounded model checking (UMC), 262–3
unique implication points (UIPs), 253, 254
unique states induction, 20
unique tables, 177–8, 205
unit clause rule, 7, 248, 252
unit clauses, 7–8
unit propagation, 248, 252
universal bus channel (UBC), 66–7
addressing, 71–4
arbitration, 69–71
data transfer, 71–4
memory access, 73–4
synchronization, 67–9
user functions, 74–5
unsatisfiability proof, 255
untimed functional model see specification model
user transaction blocks, 62
validation, IEEE definition, 124
Van Eijk’s algorithm, 24
variable reordering,
for BDDs, 4–5
for MDDs, 233
for MODDs, 223–5
for SMODDs, 228
see also dynamic reordering
variable state independent decaying sum (VSIDS)
heuristic, 255
Vera, interrupt controller case study, 160, 163–71
verification, IEEE definition, 124
verification engineers, role of, 140, 155, 160
verification plans and planning, 92–3, 111, 160
for interrupt controller, 163–4
Verilog see SystemVerilog Assertion (SVA)
package
VERTIS test generator, 145–6
VHDL hardware description language, 123
VirtexII FPGA, 85
VIS verification system, 181
Warner’s encoding, 250
Watch, 118
weighted MaxSAT, 256
weighted partial MaxSAT, 256
word-level analysis, 35
word-level ATPG, 189
word level decision diagrams (WLDDs), 182, 213
word-level equivalence checking, 189–91
word-level representation, 213
for BDDs, 182
wrappers, 60
Xilinx board, 83, 85, 88
XOR decomposition, 181–2
zero-suppressed BDD (ZBDD), 182
zero-suppressed normalized MODD (ZNMODD), 228–9