
Index

Symbols

! not operator 118
!= inequality operator 129
!== not-identity operator 130
!=? wildcard comparison 73
\$assertoff 177
\$assertvacuousoff 191
\$bitstoreal 46
\$cast 93
\$clog2 91
\$finish 171
\$readmemb() 202
\$readmemh() 202
\$realtoibits 46
\$signed 110
\$unit declarations 15, 24
\$unsigned 110
++ increment operator 112, 113, 115
+= assignment operator 112
+= assignment operators 115
.* See dot-star port connection
.name, See dot-name port connection
.per_instance coverage option 186
.randomize method 173, 175, 177, 181
.sample method 185
.sum array method 119
.sum with() array method 121
:: scope resolution operator 27
; null operation 140
<= See nonblocking assignment 62
= See blocking assignment 62
== equality operator 129

| or operator 56
=> implication operator 188
|> implication operator 188
~ invert operator 118
'timescale 195
'{ } assignment pattern operator 38, 206
'0 37
'1 37
'x 37
'z 37

Numerics

10 types of people 30
1364-2005 5, 6
1800-2005 5, 6
1-bit function return 148

A

acknowledgments ix
always 123
always_comb 51, 53, 85, 95, 97
always_ff 95, 97, 202
always_latch 51, 95, 97, 202
antecedent 188
array literals 38, 206
array method operations 119, 121
array of objects 159
assert...else 192
assertion pass statement 188
assertions 26, 79, 87, 89, 90, 155, 175, 177, 188, 189, 190, 192, 193
assign 44, 53, 95, 97, 151
assignment operators 112
assignment patterns 38, 206
assignment rules 35
assignments in expressions 99
asynchronous reset 60, 123
asynchronous set 60
automatic functions 22, 160, 161, 169

automatic package 24
 automatic programs 24, 163
 automatic tasks 22, 160, 161, 169
 automatic variables 22, 147, 162, 164

B

back-driven ports 43
 begin...end, where not to use 57, 58
 binary integer 30
 bit-select operation 111
 blocking assignment
 ++ and -- operators 112
 assignment operators 112
 correct usage 68
 definition of 62
 in clock dividers 65
 to reset 2-state models 83
 to reset at time zero 126
 boolean constraints 179

C

case 31, 76, 77, 79
 case sensitivity 7
 case statement 30
 case() inside 73
 casex 72
 casez 72
 casting 92
 casting, sign 110
 Chris Spear ix, 6
 clock dividers 64
 clocking blocks 139
 clock-to-Q delay 62, 64, 66
 coding guidelines 24, 45
 combinational logic 49, 52, 56, 57, 61, 62, 66,
 67, 68, 70, 71, 112, 116, 151
 combinational nonblocking assign 66
 compilation 41
 compilation error 8, 10, 11, 14, 17, 20, 22, 25,
 28, 29, 44, 58, 138, 139, 150, 153,
 154, 155, 159, 187
 compilation warning 14
 concatenation operator 38, 206
 concatenations 206
 concurrent for loops 145
 concurrent threads 164
 conditional compilation 61, 204, 207, 208
 consequent 188
 constructor 157, 158
 context-determined operators 101, 105, 108
 continuous assignment 44, 53, 95, 97, 151
 coverage reports 182, 184, 186
 covergroup argument direction 187

D

decimal integer 30
 decrement operator 112
 dedication v
 default direction of task argument 158
 disable 168
 disable fork 166
 Don Mills vii
 dot-name port connection 11, 14, 41
 dot-star port connection 11, 14, 41
 dynamic variables, See automatic variables

E

Emacs 11, 12, 141, 149
 enumerated types 28, 84, 92
 equality with 4-state values 129
 escaped identifiers 19, 20
 escaped names 19, 20
 event data type 131, 134
 exit simulation 171
 explicit package import 28, 29

F

FIFO 134
 filling vectors 37
 flip-flop 20, 60, 62, 64, 82, 83, 124, 202
 floating point, See real types
 for loop 142, 144, 145, 147
 foreach loop 142
 forever loop 142
 fork...join 145, 167
 fork...join_any 172
 fork...join_none 164, 172
 full_case 74, 79
 function return size 148
 functional coverage 182

G

get() method 134, 137
 get_coverage() method 182
 get_inst_coverage() method 182
 Golson, Steve ix, 1
 gotcha
 ! versus ~ 118
 \$assertoff disables randomization 177
 \$unit compilation 15
 @* 49
 1-bit implicit nets 13
 all data in mailbox has same value 157
 array literals 38
 array literals versus concatenations 206
 array of objects 159

- assert...else mismatch 192
 - assertion pass statement 188
 - assertions in procedural code 190
 - assertions that cannot fail 193
 - assignments in expressions 99
 - automatic variables 22
 - back-driven port 43
 - begin...end in sequential logic 57
 - boolean constraints 179
 - case sensitivity 7
 - casez and casex 72
 - clock dividers 64
 - combinational assignment order 70
 - combinational nonblocking assign 66
 - concurrent for loops 145
 - context-determined operators 101
 - continuous assignment with delay 151
 - coverage is 0% 184
 - coverage report lumped together 186
 - coverage reports on bins 182
 - disable fork 166
 - disabling statement blocks 168
 - enumerated types 28
 - equality with 4-state values 129
 - escaped names 19
 - event trigger races 131
 - filling vectors 37
 - function return size 148
 - functions in combinational logic 49
 - hidden problems, 2-state logic 88, 90, 92
 - hidden problems, 4-state logic 86
 - implicit nets 10
 - incomplete decisions 74
 - increment operator 112
 - infinite for loop 144
 - input versus ref arguments 158
 - literal integers 30, 32, 33, 35
 - local variables 17
 - locked state machines 84
 - mailboxes can store any data type 137
 - nested if...else blocks 128
 - non re-entrant tasks 160
 - non-standard random generator 200
 - operation short circuiting 116
 - out-of-bounds array access 90
 - out-of-bounds enumerated types 92
 - overlapped decisions 77
 - package chaining 198
 - packages 27, 28, 29
 - part-select operation 111
 - port connections 39
 - premature simulation exit 171
 - random negative values 181
 - real types on ports 46, 208
 - referencing loop variable 147
 - reset at time zero 123
 - resetting 2-state models 82
 - semaphores that don't wait 134
 - semicolons after for() 142
 - semicolons after if 140
 - sensitivity lists 52, 56
 - sequential logic 54, 64
 - sequential logic blocking assignment 62
 - sequential logic resets 59
 - sequential logic set/reset 60
 - shared variables 94, 96, 164
 - sign extension 33, 105
 - signed arithmetic 108
 - size extension 105
 - size mismatch in assignment 35
 - statements in a class 153
 - task defaults with default values 150
 - triggering on clocking blocks 139
 - undetected randomization failure 175
 - unique case misuse 79
 - unnamed blocks 25
 - using interfaces and classes 155
 - variable initialization 162
 - variables don't get randomized 173
 - variables in forked threads 164
 - zero extension 33
 - gotcha, definition of 3
 - gotcha, reasons for 4
 - gotcha, summary of gotchas in book xv
- H**
- handle 154, 155, 157, 158, 159, 173
 - hard, non-standard keyword 204
 - hex integer 30
 - hidden problems, 2-state logic 88, 90, 92
 - hidden problems, 4-state logic 86
 - hierarchical paths 19, 20, 22, 23, 25, 27
- I**
- identifiers
 - case sensitive 7
 - definition of 7
 - escaped 19
 - legal characters 19
 - IEEE SystemVerilog standard 5
 - IEEE Verilog standard 5
 - if...else 128
 - if...else, in sequential logic 57
 - implication operator 188
 - implicit nets 13
 - import, explicit 28, 29

- import, package 28, 198
 - import, wildcard 28, 29, 198
 - imported package items 27
 - incomplete decisions 74
 - increment operator 112
 - inertial delay 152
 - initial 123
 - inside 73
 - interfaces 24, 96, 155
- J**
- join_any 172
 - join_none 164, 172
- L**
- language-aware editor. 11, 128, 141, 142, 149
 - left extension, of literal value 34
 - lint checkers . . 34, 38, 44, 59, 63, 77, 94, 149
 - literal integer, size mismatch 33, 35
 - literal integers 30, 32, 33, 35
 - local variables 17, 25
 - localparam 91
 - locked simulation 66, 145
 - locked state machines 84
 - loosely typed 4, 92, 101, 107
 - LRM, SystemVerilog 5
 - LRM, Verilog 5
- M**
- mailboxes 137, 157
 - memory models 202
 - method arguments 158
 - Mills, Don vii
 - mismatch, literal value size 33, 35
 - multi-file compilation 15
 - multiple operations in one statement 115
- N**
- naming conventions 9, 24
 - negative values 181
 - negedge 54
 - nonblocking assignment
 - correct usage 63
 - definition of 62
 - exception to using 64
 - incorrect usage 66
 - intra-assignment delay 152
 - to model transport delay 152
 - to reset at time zero 126
 - nonblocking event trigger 132
 - non-standard language extensions 204
- O**
- object handles, See handle 159
 - octal integer 30
 - open-ended range in assertions 193
 - operation short circuiting 116
 - or, in sensitivity lists 56
 - out-of-bounds array access 90
 - out-of-bounds enumerated types 92
- P**
- package automatic 24
 - package, wildcard import 28, 29, 198
 - packages 16, 24, 27, 28, 29, 96
 - packages, chaining 198
 - parallel_case 74
 - parameter 91
 - part-select operation 111
 - port coercion 43
 - port connection rules 39
 - posedge 54
 - post-increment 113
 - pragmas, synthesis 74, 79
 - pre-increment 113
 - premature simulation exit 171
 - priority case 76
 - process synchronization
 - using event types 131
 - using mailboxes 137, 157
 - using semaphores 134
 - program automatic 24, 163
 - pure virtual 204
 - put() method 134
- R**
- race conditions 60, 62, 64, 83, 112, 125, 127, 131
 - rand 173
 - randc 173
 - random number generator 200
 - randomization failure 175
 - randomize 173
 - real types 46, 47, 208
 - real types on ports 46, 208
 - redundant decision selection 77
 - re-entrant tasks 160
 - ref covergroup argument 187
 - ref task/function argument 150, 158
 - reference for loop variable 147
 - repeat loop 142
 - reset at time zero 123
 - resetting 2-state models 82
 - reviewers ix

- RNG 200
rules, assignment statements 35
- S**
- scope resolution operator 27
self-determined operators 101, 105
semaphores 134
semicolons, after for() 142
semicolons, after if 140
sensitivity lists
 arrays in 52
 combinational logic 49
 function calls 49
 operations in 56
 vectors in 54
separate file compilation 15
sequential logic
 begin-end groups 57
 blocking assignments in 62
 resetting 59, 60
 sensitivity list, vectors in 54
shared declarations 15
shared variables 94, 96, 164
shift register 62
shortreal types 47, 208
sign casting 110
sign extension 33, 35, 36, 40, 102, 105
signed arithmetic rules 108
signed literal integers 32
signed types 181
signedness 32, 36, 107, 108, 110, 111
simulation lock up 66
single file compilation 15
size extension 105
solve...before 204
Spear, Chris ix, 6
state machine lock up 84
statements, in a class 153
static functions 160
static tasks 160
static variables 162
Steve Golson ix, 1
structure literals 38, 206
Stuart Sutherland vii, 6
sum array method 119
sum with() array method 121
SUPERLOG 5
Sutherland, Stuart vii, 6
synchronization 131, 134, 137, 157
synchronous reset 123
synthesis full_case pragmas 74, 79
SystemVerilog
 Accellera 3.0 standard 5
 Accellera 3.1 standard 5
 Accellera 3.1a standard 5, 206, 208
 books on 6
 definition of 3, 5
 IEEE 1800 standard 6
 SystemVerilog Assertions 87, 89, 90, 175, 177,
 188, 189, 190, 192, 193
- T**
- task and function arguments 150
time precision 195
time units 195
timeprecision 197
timeunit 197
transport delay 152
truncation
 assignments 35
 function return 148
 literal integers 33
 operations 119
 port connections 39, 144
try_get() method 137
typed mailboxes 138
- U**
- undeclared identifiers 7, 10, 13
unique case 31, 76, 77, 79
unnamed blocks 25, 26
unsigned literal integers 32
unsigned types 181
uwire 44
- V**
- vacuous success 188, 191
Value Change Dump file, See VCD
var 47, 208
var real 47, 208
variable initialization 162
variables, on ports 44
VCD 22, 26
VERA 5
Verilog
 books on 6
 definition of 3, 5
 IEEE 1364 standard 6
 LRM 5
VHDL 1, 5, 7
virtual interface 155
- W**
- wait 133, 172
wait fork 172

while loop 142
whitespace. 19, 20
wildcard comparison operator 73
wildcard package import. 28, 29, 198
wire 13, 39, 43

Z

zero extension 33, 35, 36, 40