

Index

- (see Arithmetic operators)
& (see Concatenation operator)
* (see Arithmetic operators)
** (see Arithmetic operators)
/ (see Arithmetic operators)
/= (see Comparison operators)
:= (see Assignment operators)
?/= (see Matching operators)
?< (see Matching operators)
?<= (see Matching operators)
?= (see Matching operators)
?> (see Matching operators)
?>= (see Matching operators)
| in CASE, 166
| in SELECT, 124
+ (see Arithmetic operators)
< (see Comparison operators)
<= (see Assignment operators)
<= (see Comparison operators)
= (see Comparison operators)
=> (see Assignment operators)
> (see Comparison operators)
>= (see Comparison operators)

1D array, 40, 41, 64
1D × 1D array, 41, 64, 67–69
1D × 1D × 1D array, 41, 45
2D array, 40, 41, 65, 69–70
3D array, 41, 65
4D-PAM5, 375
8B/10B, 375, 409, 410, 412

About VHDL, 3
ABS (see Arithmetic operators)
ACCESS, 245, 248
'ACTIVE (see Attributes)
Active-HDL (see EDA tools)
AD7416 temperature sensor, 421
AD7991 ADC, 421
ADC, 390, 421
Adders
ALU, 127

Carry-ripple adder, 163
Compare-add circuits, 18, 21
Fixed point, 55–57
Floating point, 57–59, 149
General, 135–140
Overloaded "+" operator, 234
Recommended fixed-point implementation, 55–57
Recommended floating-point implementation, 57–59
Recommended std-logic implementation, 139
Address decoders
Generic, 23
Generic with GENERATE, 132
AFTER, 242, 253–258, 260, 263–264
Aldec, 5
ALIAS, 112–114
ALL, 171
Allowing multiple signal assignments, 143, 190–193
Alphanumeric, LCD (see LCD)
Altera, 5, 495, 499, 500, 503, 525, 537, 545, 548
Altera DE2 board (see Tutorials)
Altera Quartus II (see Quartus II)
ALU, 127–129, 146
AND (see Logical operators)
APPEND_MODE, 246
ARCHITECTURE, 11, 16–17, 19–27
Arithmetic operators (see Operators)
ARRAY (see Data types)
Array 1D, 40, 41, 64
Array 1D × 1D, 41, 64, 67–69
Array 1D × 1D × 1D, 41, 45
Array 2D, 40, 41, 65, 69–70
Array 3D, 41, 65
Array slicing
Slicing a 1D × 1D array of bit vectors, 68
Slicing a 1D × 1D array of integers, 67
Slicing a 2D array of bits, 69
'ASCENDING (see Attributes)
ASSERT
ASSERT statement, 221–223, 238
ERROR, 222

- ASSERT (*cont.*)
 - FAILURE, 222, 223
 - NOTE, 222
 - REPORT, 221–223
 - SEVERITY, 221–223
 - WARNING, 222
- Assignment operators (*see* Operators)
- Assignment rules (*see* Objects)
- Attributes of named entities
 - E'INSTANCE_NAME, 104
 - E'PATH_NAME, 104
 - E'SIMPLE_NAME, 104
- Attributes of scalar types
 - O'SUBTYPE, 100
 - T'ASCENDING, 100
 - T'BASE, 100
 - T'HIGH, 100
 - T'IMAGE, 100, 222–223, 237
 - T'LEFT, 100
 - T'LEFTOF, 100
 - T'LOW, 100
 - T'POS, 100
 - T'PRED, 100
 - T'RIGHT, 100
 - T'RIGHTOF, 100
 - T'SUCC, 100
 - T'VAL, 100
 - T'VALUE, 100
- Attributes of signals
 - S'ACTIVE, 103
 - S'DELAYED, 103
 - S'DRIVING, 103
 - S'DRIVING_VALUE, 103
 - S'EVENT, 103
 - S'LAST_ACTIVE, 103
 - S'LAST_EVENT, 103
 - S'LAST_VALUE, 103
 - S'QUIET, 103
 - S'STABLE, 103
 - S'TRANSACTION, 103
- Attributes of synthesis
 - chip_pin, 106, 107–108
 - enum_encoding, 106–107, 279, 292
 - Keep, 106, 108–109, 140–142, 148
 - Noprune, 106, 110–111
 - Preserve, 106, 109
- Attributes, predefined, 99–104, 106–111
- Attributes, user-defined, 104–106

- 'BASE (*see* Attributes)
- BEGIN
 - In ARCHITECTURE, 16
 - In ENTITY, 15
 - In FUNCTION, 224
 - In GENERATE, 130
 - In PROCEDURE, 230
 - In PROCESS, 153
- Bibliography, 595

- Bidirectional bus
 - Implementations, 352
 - INOUT (*see* PORT modes)
 - Tri-state buffer, 50, 352
- BIT (*see* Data types)
- BIT_VECTOR (*see* Data types)
- BLOCK, 216–217
- BMP-to-RAW file converter (*see* Tutorials)
- BOOLEAN (*see* Data types)
- BOOLEAN_VECTOR (*see* Data types)
- Break code (*see* PS2)
- BUFFER (*see* PORT modes)

- Cadence, 5
- Camera link, 467
- Car alarm
 - With bypasses prevented by additional states, 296
 - With bypasses prevented by flag, 293
- Car speed monitor, 318
- CASE
 - | in CASE, 166
 - CASE statement, 151, 165–168
 - CASE? statement, 172
 - In GENERATE (*see* GENERATE statement)
 - OTHERS in CASE, 166
 - TO in CASE, 166
 - Versus SELECT, 168–169
 - WHEN in CASE, 166
- CGF (*see* Memory initialization files)
- CHARACTER (*see* Data types)
- Character representations, 8–10
- CHIP_PIN (*see* Attributes)
- Circuit with “don't care” outputs, 50
- CML, 375, 409–412
- Code structure, 11–24
- Coding guidelines, 24–27
- COE (*see* Memory initialization files)
- Combinational versus sequential circuits, 121
- Comparison operators (*see* Operators)
- COMPONENT
 - Declaration, 204–205
 - Declaration options, 205–206
 - Description, 203–213
 - File/project assembling, 205–208
 - GENERIC MAP, 18, 208–211
 - Instantiation, 204–205
 - Instantiation with GENERATE, 211–213
 - PORT MAP, 204–205, 209, 218
- Composite types (*see* Data types)
- Concatenation operator (*see* Operators)
- Concurrent statements I
 - GENERATE (*see* GENERATE)
 - SELECT (*see* SELECT)
 - WHEN (*see* WHEN)
- Concurrent statements II
 - ASSERT (*see* ASSERT)
 - BLOCK (*see* BLOCK)
 - COMPONENT inst. (*see* COMPONENT)

- PROCEDURE call (*see* PROCEDURE)
- PROCESS (*see* PROCESS)
- SIGNAL assignment (*see* Objects)
- CONFIGURATION, 26, 126, 213–216
- CONSTANT (*see* Objects)
- Conversion functions (*see* Type conversion)
 - conv_integer* (*see* Type conversion)
 - conv_signed* (*see* Type conversion)
 - conv_std_logic_vector* (*see* Type conversion)
 - conv_unsigned* (*see* Type conversion)
- CoolRunner II (*see* CPLDs)
- Counters
 - Basic, 157
 - Count leading zeros, 164
 - Gray counter, 337, 349
 - Johnson counter, 313
 - One-hot counter, 313
 - Over registered, 187
 - With SHARED VARIABLE, 179
 - With SIGNAL and VARIABLE, 183
 - Zero-to-nine, 287
 - Zero-to-nine slow with SSD, 166
- CPLDs
 - CoolRunner II, 495, 497, 499
 - CPLD devices, 495, 497, 503
 - MAX II, 495, 497, 499
 - MAX 3000, 499
 - XC9500, 499
- DAC, 421
- Data conversion (*see* Type conversion)
- Data serializer (*see* Serializer)
- Data types
 - ACCESS, 79–80
 - ARRAY, 62–70
 - Array 1D, 40, 41, 64
 - Array 1D × 1D, 41, 64, 67–69
 - Array 1D × 1D × 1D, 41, 45
 - Array 2D, 40, 41, 65, 69, 70
 - Array 3D, 41, 65
 - Array slicing, 66–70
 - Array types, 62–65
 - BIT, 42, 60
 - BIT_VECTOR, 43, 60, 76
 - BOOLEAN, 43–44, 60
 - BOOLEAN_VECTOR, 44, 60
 - CHARACTER, 46, 60
 - Composite types, 40
 - Enumerated array types, 64–65
 - Enumerated types, 61–62, 64–65
 - FILE (*see* FILE)
 - Fixed-point types, 54–57
 - FLOAT, 57–60
 - FLOAT32, 57
 - FLOAT64, 57
 - FLOAT128, 57
 - Floating-point types, 54, 57–60
 - INTEGER, 44–45, 60, 61, 63–64, 76
 - Integer array types, 63–64
 - Integer types, 61
 - INTEGER_VECTOR, 45, 60
 - Integer versus enumerated indexing, 65–66
 - NATURAL, 45, 60
 - POSITIVE, 45, 60
 - Predefined types, 41–60
 - Qualified types, 73–74
 - REAL, 46
 - REAL_VECTOR, 46
 - RECORD, 70–71
 - Scalar types, 61–62
 - SFIXED, 54–57, 60
 - SIGNED, 51–54, 60, 128, 135–140
 - Signed types, 51–54, 60, 76
 - Standard-logic types, 47–51, 60, 76
 - Standard types, 41–47, 60, 76
 - STD_LOGIC, 37, 47–51, 60
 - STD_LOGIC_VECTOR, 37, 47–51, 60, 76
 - STD_ULOGIC, 37, 47–51, 60
 - STD_ULOGIC values ('U', 'X', '0', '1', 'Z', 'W', 'L', 'H', '-'), 47
 - STD_ULOGIC_VECTOR, 37, 47–51, 60
 - STRING, 46, 60, 222–223, 237, 245–249
 - SUBTYPE, 71–72
 - Summary, 60
 - TIME, 242, 245, 246, 247
 - TIME_VECTOR, 47
 - TYPE, 61–66
 - Type classifications, 39–41
 - Type conversion, 74–78
 - UFIXED, 55–57, 60, 76
 - UNSIGNED, 51–54, 60, 128, 135–140
 - Unsigned types, 51–54, 60
 - User-defined array types, 62–65
 - User-defined enumerated array types, 64–65
 - User-defined integer array types, 63–64
 - User-defined scalar types, 60–62
 - User-defined types, 60–70
- DDC (*see* VGA and DVI)
- DDWG, 445
- DE2 board (*see* Tutorials)
- Dealing with files (*see* Simulation)
- Debouncer, 174
- Deferred CONSTANT (*see* Objects)
- 'DELAYED' (*see* Attributes)
- Delay line
 - With KEEP, 108, 140
 - With LCELL, 142
- Deserializer (*see* Serializer)
- Design Compiler Ultra (*see* EDA tools)
- Design/simulation flow, 5
- Design of state machines (*see* FSM)
- Development board (*see* DE2 board)
- DFF (*see* Flip-flops)
- Digital (wall) clock
 - With LCD screen, 479
 - With SSDs, 337

- DIN connector, 381
- Disparity, 412–417
- Display resolutions
 - Full HD, 448
 - QSXGA, 448
 - QUXGA, 448
 - QXGA, 448
 - SVGA, 423–424, 441, 448
 - SXGA, 423–424, 448
 - UXGA, 448
 - VGA, 423–424, 448
 - WQUXGA, 448
 - WUXGA, 448
 - XGA, 423–424, 448
- Divider
 - Division operators (*see* Operators)
 - Floating point, 149
- Don't care, 47, 50–51, 144, 172
- DOWNTO/TO, 43
- 'DRIVING' (*see* Attributes)
- 'DRIVING_VALUE' (*see* Attributes)
- DS1306 RTC, 422
- DS3231 RTC, 421
- D-type flip-flop (*see* Flip-flops)
- Dual-edge DFF (*see* Flip-flops)
- DVI
 - Circuit diagram, 446–448
 - DDC and EDID, 446–448, 450
 - Description, 445–465
 - Display resolutions, 448
 - DVI connectors, 449–450
 - DVI types, 449–450
 - HDMI, 450–451
 - HPD, 448, 450
 - Image generation with file and off-chip memory, 438–441, 444
 - Image generation with file and on-chip memory, 435–438, 444
 - Image generation with hardware, 431–435, 442–443
 - Image scaler, 448
 - PLL, 447, 454–455, 457
 - Serializer, 447, 460–461
 - Setup for the experiments, 451–452
 - TMDS encoder, 457–460
- EDA tools
 - Active-HDL, 5, 242
 - Altera Quartus II (*see* Quartus II)
 - Design Compiler Ultra, 5
 - ISE/XST, 5, 241, 515–523
 - ISE/XST tutorial (*see* Tutorials)
 - Leonardo Spectrum, 5
 - ModelSim, 5, 241, 525–535
 - ModelSim tutorial (*see* Tutorials)
 - NC-Sim, 5, 242
 - Precision RTL, 5
 - Quartus II, 5, 241, 503–514
 - Quartus II tutorial (*see* Tutorials)
 - Synplify Pro/Premier, 5
 - VCS, 5, 241
 - Xilinx ISE (*see* ISE/XST)
- EDID (*see* VGA and DVI)
- EEPROM memory (*see* I²C)
- 'ELEMENT' (*see* Attributes)
- ELSE
 - In GENERATE, 144
 - In IF, 155
 - In WHEN, 123
- ELSIF
 - In GENERATE, 144
 - In IF, 155
- Encoding styles (*see* FSM encoding)
- END
 - In ARCHITECTURE, 16
 - In CASE, 166
 - In COMPONENT, 204
 - In CONFIGURATION, 214
 - In ENTITY, 14
 - In FUNCTION, 224
 - In GENERATE, 130
 - In IF, 155
 - In LOOP, 161
 - In PACKAGE/BODY, 201
 - In PROCEDURE, 230
 - In PROCESS, 153
- ENDFILE, 248–250, 270
- ENTITY, 11, 14–16, 19–25
- ENUM_ENCODING (*see* Attributes of synthesis)
- Enumerated types (*see* Data types)
- env* (*see* Packages)
- ERROR (*see* ASSERT)
- 'EVENT' (*see* Attributes)
- Examples
 - 2.1 Compare-add circuit, 18
 - 2.2 D-type flip-flop (DFF), 20
 - 2.3 Registered comp-add circuit, 21
 - 2.4 Generic address decoder, 23
 - 3.1 Tri-state buffer, 50
 - 3.2 Circuit with 'don't care' outputs, 50
 - 3.3 Unsigned/signed multiplier #1, 52
 - 3.4 Unsigned/signed multiplier #2, 54
 - 3.5 Slicing a 1D × 1D array of integers, 67
 - 3.6 Slicing a 1D × 1D array of bit vectors, 68
 - 3.7 Slicing a 2D array of bits, 69
 - 3.8 Multiplexer with 1D × 1D PORT, 72
 - 3.9 Recommended signed multiplier implementation (for integers), 75
 - 3.10 Legal versus illegal assignments, 79
 - 4.1 Using Predefined Scalar Attributes, 101
 - 4.2 DFF with several event-based attributes, 103
 - 4.3 Specifying device pins with the *chip_pin* attribute, 107
 - 4.4 Construction of a delay line with the *keep* attribute, 108
 - 4.5 Keeping redundant registers with *preserve* and *noprune* attributes, 110

- 5.1 Multiplexer implemented with operators, 122
- 5.2 Multiplexer implemented with WHEN and SELECT, 125
- 5.3 ALU, 127
- 5.4 Generic address decoder with GENERATE, 132
- 5.5 COMPONENT instantiation with GENERATE, 133
- 5.6 DFF implemented with concurrent code, 134
- 5.7 Recommended adder/subtractor implementation, 139
- 5.8 Short-pulse generator with the *keep* attribute, 140
- 5.9 Short-pulse generator with the LCELL primitive, 142
- 6.1 DFFs with reset and clear, 155
- 6.2 Basic counter, 157
- 6.3 Shift register, 158
- 6.4 Carry-ripple adder, 163
- 6.5 Leading zeros, 164
- 6.6 Slow 0-to-9 counter with SSD, 166
- 6.7 Incomplete combinational design, 169
- 7.1 Counter with SHARED VARIABLE, 179
- 7.2 SIGNAL versus VARIABLE usage, 182
- 7.3 Counters with SIGNAL and VARIABLE, 183
- 7.4 DFF with *q* and *qbar*, 186
- 7.5 Over-registered counter, 187
- 7.6 Dual-edge flip-flop, 189
- 7.7 Generic Hamming weight with concurrent code, 192
- 8.1 PACKAGE with FUNCTION and deferred CONSTANT, 203
- 8.2 Circular shift register with COMPONENT, 206
- 8.3 Parity detector with COMPONENT and GENERIC MAP, 209
- 8.4 Shift register with COMPONENT and GENERATE, 211
- 8.5 Latch implemented with a guarded BLOCK, 217
- 9.1 Function *max* in ARCHITECTURE, 226
- 9.2 Function *order_and_fill* in PACKAGE, 227
- 9.3 Function *slv_to_integer* in ENTITY, 229
- 9.4 Procedure *min_max* in PACKAGE, 231
- 9.5 Overloaded "+" operator, 234
- 9.6 Non-overloaded "AND" operator, 235
- 10.1 Writing values to a file, 236
- 10.2 Reading values from a file, 248
- 10.3 Stimuli generation, 256
- 10.4 Type I testbench for a registered mux, 259
- 10.5 Type II testbench for a registered mux, 261
- 10.6 Type IV testbench for a registered mux, 262
- 10.7 Type IV testbench with a record type, 264
- 10.8 Type IV testbench with a data file, 268
- 11.1 Vending-machine controller, 281
- 11.2 Glitch-free vending-machine controller, 286
- 11.3 Zero-to-nine counter, 287
- 11.4 Car alarm with bypasses prevented by a flag, 293
- 11.5 Car alarm with bypasses prevented by additional states, 296
- 11.6 FSM with embedded timer, 301
- 11.7 Traffic-light controller, 303
- 11.8 Pushbutton sequence detector, 307
- 12.2 Basic LED/SSD/LCD driver, 322
- 12.3 Playing with a seven-segment display, 327
- 12.4 Frequency meter (with LCD), 330
- 12.5 Digital clock (with SSDs), 337
- 12.6 Quick-finger game (with LEDs and SSDs), 340
- 12.7 Other designs with basic displays, 345
- 13.2 Implementing bidirectional buses, 352
- 13.4a ROM implemented with regular VHDL code, 358
- 13.4b ROM implemented with an initialization file, 360
- 13.4c ROM implemented with a vendor-specific function, 361
- 13.5a RAM implemented with regular VHDL code, 363
- 13.5b RAM implemented with a vendor-specific function, 365
- 13.5c RAM implemented in a user SRAM block, 366
- 13.6 External memory interfaces, 368
- 14.2a Design of a fast serializer, 379
- 14.3a Design of a PS2 keyboard interface, 384
- 14.4a Design of an I²C interface for an EEPROM memory, 390
- 14.5a Design of an SPI interface for an FRAM memory, 402
- 14.6a Design of a TMDS encoder, 414
- 15.9 Hardware-Generated Image, 431
- 15.10 Image Generation with a File and On-Chip Memory, 435
- 15.11 Arbitrary Image Generation with a File and Off-Chip Memory, 438
- 15.12 Image Equalization with Gamma Expansion, 441
- 16.7 Hardware-Generated Image, 452
- 17.4 Hardware-Generated Image, 472
- 17.5 Hardware-Generated Image with Characters, 479
- EXIT (*see* LOOP)
- FAILURE (*see* ASSERT)
- FILE, 35, 80, 245–249
- FILE_OPEN_KIND, 246
- FILE_OPEN_STATUS, 246
- Finite state machines (*see* FSM)
- fixed_float_types* (*see* Packages)
- fixed_generic_pkg* (*see* Packages)
- fixed_pkg* (*see* Packages)
- Fixed-point adder/multiplier, 56–57
- Fixed-point types (*see* Data types)
- Flash memory, 421
- Flip-flops
 - Basic DFF, 20–21
 - DFF, 20–22, 103–104, 110, 134–135, 152–153, 155–160, 181–182, 185–190, 206–208, 211–213, 278, 313–315
 - DFF with clear, 153, 160

- Flip-flops (*cont.*)
 - DFF with q and $qbar$, 186
 - DFF with reset, 153
 - DFF with several event-based attributes, 103–104
 - Dual-edge, 187–190
 - From concurrent code, 134–135
 - Inference of flip-flops, 181–182, 185–188, 313–315
 - Keeping redundant registers with PRESERVE and NOPRUNE, 110
 - Number of flip-flops, 281, 292, 304, 313, 317, 313–315
 - Setup/hold times, 330, 336
- FLOAT (*see* Data types)
- float_generic_pkg (*see* Packages)
- Floating-point adder/multiplier, 57–59
- Floating-point adder/subtractor/multiplier/divider, 150
- Floating-point types (*see* Data types)
- float_pkg (*see* Packages)
- FOR
 - In CONFIGURATION, 214
 - In GENERATE, 130
 - In LOOP, 161
 - In WAIT, 159
- FPD-Link
 - Description, 467–492
 - Digital clock on LCD screen, 479–490
 - Display LB104S01, 468
 - Display resolutions, 424, 448
 - Display SFA-104A, 468
 - FPD-Link encoder, 468–470
 - Image generation with file and off-chip memory, 492
 - Image generation with file and on-chip memory, 491–492
 - Image generation with hardware, 472–490, 491
 - Image with characters, 479–490
 - LCD display, 467–468
 - LDI, 467
 - LVDS, 467, 469, 471
 - Native display resolution, 468
 - PLL, 474–476
 - Serializer, 470, 476
 - Setup for the experiments, 470–471
- FPGAs
 - ALM, 499
 - CLB, 499
 - FPGA devices, 496, 499–501, 503, 538
 - LAB, 499
 - Slice, 499
 - Stratix IV, 500
 - Virtex, 6 500
- FRAM memory (*see* SPI)
- Frequency divider, 197
- Frequency meter, 175, 220, 330–337, 347
- FSM (*see* specific FSM field)
- FSM design
 - Design of basic FSMs, 279–289
 - Design of complex/timed FSMs, 292–298, 298–312
 - Encoding styles (*see* FSM encoding)
 - FSM block diagram, 278
 - FSMs with repetitive states, 312
 - Glitch-free FSMs, 278–279, 281–282, 286, 288, 314, 315
 - Golden design rules, 281
 - Hardware representation (*see* FSM models)
 - Models (*see* FSM models)
 - State-bypass problem in FSMs, 292–298, 301, 308, 315, 316
 - State transition diagram (*see* FSM models)
 - VHDL template for basic FSMs, 279–280
 - VHDL template for complex/timed FSMs, 299–300
- FSM encoding
 - ENUM_ENCODING (*see* Attributes of synthesis)
 - Gray encoding, 291–292, 313
 - Johnson encoding, 291, 313
 - One-hot encoding, 291–292, 304, 313, 314, 317
 - Sequential encoding, 291–292, 294, 296, 301, 304, 309
 - User-defined encoding, 292
- FSM models
 - Conditional-only FSMs, 298–299
 - Conditional-timed FSMs, 298–299
 - Model for the hardware, 277–278
 - Poor FSM model, 289–291
 - State transition diagram, 277–278, 282, 287
 - Timed-only FSMs, 298–299
- Full HD (*see* Display resolutions)
- FUNCTION
 - Call, 225
 - Construction, 224–225, 238–239
 - Description, 223–230, 237
 - Function *slv_to_integer*, 229–230
 - IMPURE, 224, 226
 - In ARCHITECTURE, 226–227
 - In ENTITY, 229–230
 - In PACKAGE, 227–229
 - integer_to_lcd*, 323–325, 331, 333
 - integer_to_ssd*, 323–325, 337–340
 - max*, 226–227
 - Non-overloaded "AND" operator, 235–237
 - order_and_fill, 227–229
 - Overloaded "+" operator, 234–235
 - Overloading, 233–237
 - PURE, 224, 226
 - RETURN, 224
 - Versus PROCEDURE, 233
- Fundamental VHDL units, 11–12
- GAL (*see* SPLDs)
- GENERATE
 - CASE in GENERATE, 144
 - Conditional, 130
 - ELSE/ELSIF in GENERATE, 144
 - FOR, 130
 - GENERATE statement, 129–134, 144, 146
 - IF in GENERATE, 130, 144
 - IN in GENERATE, 130
 - Unconditional, 130

- GENERIC, 15–16, 17–18, 25–28, 132, 204, 209
- GENERIC MAP (*see* COMPONENT)
- Geometric game clock, 341
- Glitch-free state machines (*see* FSM design)
- Gray code (*see* FSM encoding)
- Gray counter, 337, 349–350
- GROUP, 111–112
- Guarded BLOCK, 217
- Guarded SIGNAL (*see* Objects)
- Hamming weight calculator, 192
- HD44780U microcontroller, 320
- HDMI (*see* DVI)
- HEX (*see* Memory initialization files)
- 'HIGH (*see* Attributes)
- HPD (*see* DVI)
- I²C
 - Description, 388–399
 - EEPROM AT24C02B, 390
 - Interface design for EEPROM, 390–399
 - Interface for ADC, 421
 - Interface for RTC, 421
 - Interface for temperature sensor, 421
- IEEE library, 12–13
- IF
 - ELSE/ELSIF, 155
 - IF statement, 151, 154–159
 - In GENERATE (*see* GENERATE statement)
 - THEN, 155
- Illegal assignments, 78–79
- 'IMAGE (*see* Attributes)
- Image generators
 - For DVI (*see* DVI)
 - For FPD-Link (*see* FPD-Link)
 - For VGA (*see* VGA)
- Image scaler (*see* DVI)
- Implementing arithmetic circuits with operators, 135–140
- Implementing bidirectional buses, 352
- Implementing combinational circuits with sequential code, 169–171
- Implementing sequential circuits with concurrent code, 134–135
- IMPURE (*see* FUNCTION)
- IN
 - In GENERATE, 130
 - In LOOP, 161
 - In PORT (*see* PORT modes)
 - In subprograms (*see* PORT modes)
- Inference of flip-flops (*see* Flip-flops)
- Inference of registers (*see* Flip-flops)
- INOUT (*see* PORT modes)
- 'INSTANCE_NAME (*see* Attributes)
- INTEGER (*see* Data types)
- integer_to_lcd (*see* FUNCTION)
- integer_to_ssd (*see* FUNCTION)
- Integer types (*see* Data types)
- INTEGER_VECTOR (*see* Data types)
- IPCA8563 RTC, 421
- IS
 - In ARCHITECTURE, 16
 - In COMPONENT, 204
 - In CONFIGURATION, 214
 - In ENTITY, 14
 - In FUNCTION, 224
 - In PACKAGE (BODY), 201
 - In PROCEDURE, 230
 - In PROCESS, 154
- ISE/XST (*see* EDA tools)
- ISE/XST tutorial (*see* Tutorials)
- Johnson code (*see* FSM encoding)
- Johnson counter, 313
- KEEP (*see* Attributes of synthesis)
- Keyboard encoding (*see* PS2)
- 'LAST_ACTIVE (*see* Attributes)
- 'LAST_EVENT (*see* Attributes)
- 'LAST_VALUE (*see* Attributes)
- Latches, 152–153, 166, 169, 172
- LCD alphanumeric
 - Alphanumeric display, 320–321
 - Frequency meter with LCD, 330–337
 - HD44780U microcontroller, 320
 - LCD, 319–327, 330–335, 346–348
 - LCD driver, 322–327
- LCD screen
 - Designs with LCD screen (*see* DVI and FPD-Link)
 - Digital clock on LCD screen (*see* FPD-Link)
 - LB104S01 display, 468
 - LCD monitor, 445–446, 467–468
 - Monitor resolutions (*see* Display resolutions)
 - Native display resolution, 468
 - SFA-104A display, 468
- LCELL primitive, 142
- LDI (*see* FPD-Link)
- LED
 - Description, 319–320
 - LED driver, 322
 - Pushbutton sequence detector (with LED), 307–311
 - Quick-finger game (with LED), 340–345
 - Traffic-light controller (with LED), 303
- 'LEFT (*see* Attributes)
- 'LEFTOF (*see* Attributes)
- Legal assignments, 78–79
- Legal versus illegal assignments, 78–79
- 'LENGTH (*see* Attributes)
- Leonardo Spectrum (*see* EDA tools)
- Libraries
 - ieee, 12–13
 - std, 12
 - work, 13
- Library declarations, 13–14
- Light emitting diode (*see* LED)

- LINE, 245–250
- Line codes, 375, 409
- Liquid crystal display (*see* LCD)
- LM75A temperature sensor, 421
- Logical operators (*see* Operators)
- LOOP
 - LOOP statement, 151, 161–165
 - LOOP with EXIT, 161–162, 165
 - LOOP with FOR, 161–164
 - LOOP with NEXT, 162–163
 - LOOP with WHILE, 161–162, 172
 - Unconditional, 162
- 'LOW (*see* Attributes)
- Low-pixel format, 448
- LVC MOS, 375, 411, 452
- LVDS, 375, 467, 469, 471

- Macrofunctions (*see* Tutorials)
- Main properties of SIGNAL, 151–152
- Main properties of VARIABLE, 152
- Make code (*see* PS2)
- Making multiple signal assignments, 190–193
- Matching operators (*see* Operators)
- MAX II (*see* CPLDs)
- MAX1242 ADC, 421
- MAX 3000 (*see* CPLDs)
- MAXIMUM (*see* Operators)
- Memory
 - EEPROM (*see* I²C)
 - Flash, 421
 - FRAM (*see* SPI)
 - RAM (*see* RAM)
 - ROM (*see* ROM)
 - SRAM (*see* RAM)
- Memory initialization files
 - CGF file, 354–355, 372
 - COE file, 354–355, 372
 - HEX file, 354, 356, 360–361, 372
 - MIF file, 354, 360–362, 372
- Mentor Graphics, 5, 525
- Metastable states, 330, 336
- MIF (*see* Memory initialization files)
- MINIMUM (*see* Operators)
- MLT-3, 375
- MOD (*see* Arithmetic operators)
- ModelSim (*see* EDA tools)
- ModelSim tutorial (*see* Tutorials)
- Mouse encoding (*see* PS2)
- Multiple ENTITY-ARCHITECTURE pairs (*see* CONFIGURATION)
- Multiple signal assignments, 143, 190–193
- Multiplexers
 - With 1D × 1D PORT, 72
 - With COMPONENT and GENERATE, 133
 - With operators, 122
 - With WHEN and SELECT, 125
- Multipliers
 - Fixed point, 55–57
 - Floating point, 57–59, 150
 - Recommended signed, 75, 148
 - Recommended unsigned, 148
 - Unsigned/signed, 52–54, 75, 148
- NAND (*see* Logical operators)
- Native display resolution, 468
- NATURAL (*see* Data types)
- NC-Sim (*see* EDA tools)
- NEW, 80, 218
- NEXT (*see* LOOP)
- Non-overloaded AND operator (*see* FUNCTION)
- NO PRUNE (*see* Attributes)
- NOR (*see* Logical operators)
- NOT (*see* Logical operators)
- NOTE (*see* ASSERT)
- NOW, 242, 253, 264
- NULL, 80, 166, 169
- Number of flip-flops (*see* Flip-flops)
- Number representations, 8–10
- numeric_bit (*see* Packages)
- numeric_bit_unsigned (*see* Packages)
- numeric_std (*see* Packages)
- numeric_std_unsigned (*see* Packages)

- Objects
 - Assignment operators (*see* Operators)
 - Assignment rules, 78–79
 - CONSTANT, 32
 - Deferred CONSTANT, 32–33, 203
 - FILE (*see* FILE)
 - Guarded SIGNAL, 178
 - Inference of registers, 181–182, 185–190
 - Legal versus illegal assignments, 78–79
 - Multiple signal assignments, 190–193
 - Properties of SIGNAL, 151–152, 180–182
 - Properties of VARIABLE, 152, 180–182
 - Resolved SIGNAL, 178
 - SHARED VARIABLE, 179–180
 - SIGNAL, 33–34, 151–152, 177–178, 180–190
 - SIGNAL versus VARIABLE, 180–190
 - VARIABLE, 34–35, 152, 178–190
- OF
 - In ARCHITECTURE, 16
 - In ARRAY, 43–48, 62–63
 - In ATTRIBUTE, 104–107
 - In CONFIGURATION, 214
 - In FILE, 35
- ON (*see* WAIT)
- One-hot code (*see* FSM encoding)
- One-hot counter, 313
- Operators
 - Arithmetic operators (+, −, *, /, **, ABS, MOD, REM), 93–94, 114
 - Assignment operators (<=, :=, >=), 92
 - Comparison operators (=, /=, <, >, <=, >=), 94, 114
 - Concatenation operator (&), 96

- Logical operators (NOT, AND, NAND, OR, NOR, XOR, XNOR), 92–93, 114
- Matching operators (?=, ?/=, ?<, ?>, ?<=, ?>=), 96–97, 114
- MAXIMUM, 97, 115
- MINIMUM, 97, 115
- Overloaded operators, 74, 98, 233–237
- Overloading, 74, 98, 233–237
- Predefined operators, 91–98
- Shift operators (SLL, SRL, SLA, SRA, ROL, ROR), 95, 114
- Summary, 97–98
- TO_HSTRING, 97, 115
- TO_OSTRING, 97, 115
- TO_STRING, 97, 115
- User-defined operators, 98
- Using operators, 122–123
- OR (see Logical operators)
- OTHERS
 - Description, 33, 92, 96, 124, 166
 - In CASE, 166
 - In SELECT, 124
- OUT (see PORT modes)
- Overloaded "+" operator (see FUNCTION)
- Overloaded operators (see Operators)
- Overloading (see Operators)
- Over-registered counter, 187
- PACKAGE
 - Declarations, 13–14
 - Description, 201–203, 206, 212, 218
 - With FUNCTION, 323–324
- PACKAGE BODY, 201–203, 218
- Packages
 - env*, 12
 - fixed_float_types*, 36, 55, 57, 58
 - fixed_generic_pkg*, 36, 55
 - fixed_pkg*, 12, 36, 39, 55, 58, 60
 - fixed_pkg* compatible, 55
 - float_generic_pkg*, 36, 57
 - float_pkg*, 13, 36, 39, 57, 58, 60
 - float_pkg* compatible, 58
 - numeric_bit*, 12, 36, 60
 - numeric_bit_unsigned*, 12, 36, 39, 60
 - numeric_std*, 12, 36, 38, 60, 563–576
 - numeric_std_unsigned*, 12, 36, 39, 60, 589–592
 - standard*, 12, 36–37, 60, 551–554
 - std_logic_1164*, 12, 36, 37, 60, 555–562
 - std_logic_arith*, 13, 36, 38, 60, 577–581
 - std_logic_signed*, 13, 36, 38, 60, 583–584
 - std_logic_unsigned*, 13, 36, 38, 60
 - textio*, 12, 36, 39, 585–587
- PAL (see SPLDs)
- Parity bit, 381, 383
- Parity detector, 209
- Parity generator, 145
- 'PATH_NAME (see Attributes)
- PCF8591 ADC, 421
- PECL, 375
- PLA (see SPLDs)
- Playing with an SSD, 327–330
- PLL, 376–380, 447, 454, 474, 547–550
- PORT, 14–16, 19–25
- PORT arrays, 72–73
- PORT MAP (see COMPONENT)
- PORT modes
 - BUFFER, 14–15
 - IN, 14–15
 - INOUT, 14–15
 - OUT, 14–15
- 'POS (see Attributes)
- POSITIVE (see Data types)
- Precision RTL (see EDA tools)
- 'PRED (see Attributes)
- Predefined attributes, 99–104, 106–111
- Predefined operators (see Operators)
- Predefined types (see Data types)
- PRESERVE (see Attributes)
- Preventing combinational-logic simplification, 140–143
- PROCEDURE
 - Description, 230–233, 237–238
 - In PACKAGE, 223–231
 - min_max*, 231–233
 - Versus FUNCTION, 233
- PROCESS, 151, 153–154, 156–161, 164–171
- Properties of SIGNAL (see Objects)
- Properties of VARIABLE (see Objects)
- PS2
 - Break code, 383, 385
 - Design of keyboard interface, 384–388
 - Keyboard encoding, 383
 - Make code, 381, 383, 385, 387
 - Mouse encoding, 383–384
 - PS2 interface, 380–388, 420
 - Scan code set, 383
- PURE (see FUNCTION)
- Pushbutton sequence detector, 307, 317
- QSXGA (see Display resolutions)
- Qualified types (see Data types)
- Quartus II (see EDA tools)
- Quartus II tutorial (see Tutorials)
- Quick-finger game (with LEDs/SSDs), 340–345, 347
- 'QUIET (see Attributes)
- QUXGA (see Display resolutions)
- QXGA (see Display resolutions)
- RAM
 - Description, 351–352, 362–371
 - Implemented in user SRAM blocks, 366–368
 - Implemented with regular code, 363–365
 - Implemented with vendor-specific function, 365–366
 - IS61LV25616 SRAM, 368
 - SRAM blocks, 351, 363, 365–368
 - SRAM interface, 368–371

- 'RANGE (*see* Attributes)
- READ, 248–250, 268, 270
- Reading files (*see* Simulation)
- READ_MODE, 246, 248, 270
- REAL (*see* Data types)
- Real time clock (*see* RTC)
- REAL_VECTOR (*see* Data types)
- RECORD (*see* Data types)
- REM (*see* Arithmetic operators)
- REPORT (*see* ASSERT)
- Representations for numbers and characters, 8–10
- Reserved VHDL words, 593
- Resolved SIGNAL (*see* Objects)
- RETURN (*see* FUNCTION)
- 'REVERSE_RANGE (*see* Attributes)
- 'RIGHT (*see* Attributes)
- 'RIGHTOF (*see* Attributes)
- ROL (*see* Shift operators)
- ROM
 - Description, 351–352, 354, 357–362, 370, 372
 - Implemented with initialization file, 360–361
 - Implemented with regular code, 358–360
 - Implemented with vendor-specific function, 361–362
- ROR (*see* Shift operators)
- RTC, 388, 400, 401, 421
- Running disparity, 412–417

- Scalar types (*see* Data types)
- Scan code set (*see* PS2)
- SDF file, 241, 245, 253, 261, 267
- SELECT
 - | in SELECT, 124
 - OTHERS in SELECT, 124
 - SELECT statement, 124–129, 143, 168–169
 - SELECT? statement, 143
 - TO in SELECT, 124
 - Versus CASE, 168–169
 - WHEN in SELECT, 124
 - WITH in SELECT, 124
- Sensitivity list (*see* PROCESS)
- Sequential encoding (*see* FSM encoding)
- Serializers
 - Data deserializers, 378, 420
 - Data serializers, 376–380, 419–420, 447, 460, 470, 476
 - Deserializer circuit, 378
 - Design of fast serializer, 379–380
 - Serializer circuits, 376–380
- Setup/hold times (*see* Flip-flops)
- Seven segment display (*see* SSD)
- SEVERITY (*see* ASSERT)
- SFIXED (*see* Data types)
- SHARED_VARIABLE (*see* Objects)
- Shift operators (*see* Operators)
- Shift registers
 - Basic, 158
 - With COMPONENT, 206
 - With COMPONENT and GENERATE, 211
- Short-pulse generators
 - With KEEP, 140
 - With LCELL, 142
- SIDE, 245, 248
- SIGNAL (*see* Objects)
- Signal generator, 174, 175, 176, 220, 314, 315
- SIGNAL versus VARIABLE (*see* Objects)
- SIGNED (*see* Data types)
- signed* (*see* Type conversion)
- Signed types (*see* Data types)
- 'SIMPLE_NAME (*see* Attributes)
- Simulation
 - Automated functional simulation (*see* Type III testbench)
 - Automated timing simulation (*see* Type IV testbench)
 - Dealing with files, 245–250
 - Design file (for simulation), 244–245, 259, 265, 268
 - Full-bench (*see* Type IV testbench)
 - Functional simulation, 252
 - Graphical simulation, 243–244, 251–253
 - Manual functional simulation (*see* Type I testbench)
 - Manual timing simulation (*see* Type II testbench)
 - Postsynthesis file, 245
 - Reading data from files, 248–250
 - RTL, 4, 241, 242
 - SDF file (*see* SDF)
 - Simulation with testbenches, 241–275
 - Stimulus generation, 253–257
 - Template for testbenches, 257–258
 - Testbenches with data files, 267–270
 - Testbenches with record types, 264–267
 - Test file (for simulation), 244, 245, 259, 262, 264–265, 268
 - Timing simulation, 253
 - Tools (*see* EDA tools)
 - Type I testbenches, 243–245, 258–261
 - Type II testbenches, 243–245, 261
 - Type III testbenches, 243–245, 261–262
 - Type IV testbenches, 243–245, 262–270
 - Writing data to files, 245–247
- SLA (*see* Shift operators)
- SLL (*see* Shift operators)
- Softwares (*see* EDA tools)
- SPI
 - Description, 399–409
 - FRAM FM25L512, 401
 - Interface design for FRAM, 402–409
 - Interface for ADC, 421
 - Interface for flash memory, 421
 - Interface for FRAM, 421
 - Interface for RTC, 422
- SPLDs
 - GAL devices, 495, 497
 - PAL devices, 495–497
 - PLA devices, 495–497
 - SPLD devices, 495–496
- SRA (*see* Shift operators)

- SRAM (*see* RAM memory)
- SRL (*see* Shift operators)
- SSD
 - Description, 319–320
 - Digital clock (with SSD), 337–340
 - One-digit timer (with SSD), 166–168
 - Playing with an SSD, 327–330
 - Quick finger game (with SSD), 340–345
 - SSD driver, 322
 - Two-digit timer (with SSD), 174
- 'STABLE (*see* Attributes)
- standard* (*see* Packages)
- Standard-logic types (*see* Data types)
- Standard types (*see* Data types)
- Start bit, 375, 381–385
- State machines (*see* FSM)
- State transition diagram (*see* FSM models)
- std* library, 12
- STD_LOGIC (*see* Data types)
- std_logic_1164 (*see* Packages)
- std_logic_arith (*see* Packages)
- std_logic_signed (*see* Packages)
- std_logic_unsigned (*see* Packages)
- STD_LOGIC_VECTOR (*see* Data types)
- std_logic_vector* (*see* Type conversion)
- STD_ULOGIC (*see* Data types)
- STD_ULOGIC_VECTOR (*see* Data types)
- Stimulus generation (*see* Simulation)
- Stop bit, 375, 381–385
- Stratix IV (*see* FPGAs)
- STRING (*see* Data types)
- Subtractor (*see* Adders)
- SUBTYPE (*see* Data types)
- 'SUBTYPE (*see* Attributes)
- 'SUCC (*see* Attributes)
- SVGA (*see* Display resolutions)
- SXGA (*see* Display resolutions)
- Synchronizer, 330, 336, 349
- Synopsys/Synplicity, 5
- Synplify Pro/Premier (*see* EDA tools)
- Synthesis attributes (*see* Attributes)
- Synthesis tools (*see* EDA tools)

- Tapped delay line, 220
- Temperature sensor, 388, 421
- Templates for state machines (*see* FSM design)
- Testbenches (*see* Simulation)
- TEXT, 245–249, 270
- textio* (*see* Packages)
- THEN (*see* IF)
- TIME (*see* Data types)
- Timer, 166, 174, 347–348
- TIME_VECTOR (*see* Data types)
- TMDS
 - Description, 409–419, 422
 - Disparity, 412–417
 - Encoding algorithm, 412–414
 - Implementation, 414–419

- TO
 - In CASE, 166
 - In GENERATE, 131
 - In LOOP, 162
 - In SELECT, 124
 - In type ranges (TO/DOWNTO), 32, 33, 44
- to_bitvector* (*see* Type conversion)
- to_float* (*see* Type conversion)
- TO_HSTRING (*see* Operators)
- to_integer* (*see* Type conversion)
- Tools (*see* EDA tools)
- TO_OSTRING (*see* Operators)
- to_sfixed* (*see* Type conversion)
- to_signed* (*see* Type conversion)
- to_slv* (*see* Type conversion)
- to_stdlogicvector* (*see* Type conversion)
- TO_STRING (*see* Operators)
- to_ufixed* (*see* Type conversion)
- to_unsigned* (*see* Type conversion)
- Traffic-light controller, 303
- 'TRANSACTION (*see* Attributes)
- Tri-state buffer
 - Basic, 50
 - Bidirectional bus, 352
 - INOUT (*see* PORT modes)
- TTL 375
- Tutorials
 - BMP-to-RAW file converter, 545–546
 - DE2 board, 537–543
 - ISE/XST, 515–523
 - Macrofunctions, 547–550
 - ModelSim, 525–535
 - Quartus II, 503–514
- TYPE (*see* Data types)
- Type casting (*see* Type conversion)
- Type classifications (*see* Data types)
- Type conversion
 - Automatic conversion, 75
 - conv_integer, 76
 - conv_signed, 76
 - conv_std_logic_vector, 76
 - conv_unsigned, 76
 - signed, 75–78
 - std_logic_vector, 75–78
 - Summary, 76
 - to_bitvector, 76
 - to_float, 76
 - to_integer, 76
 - to_sfixed, 76
 - to_signed, 76
 - to_slv, 76
 - to_stdlogicvector, 76
 - to_ufixed, 76
 - to_unsigned, 76
 - Type casting, 75
 - unsigned, 76
- Type I testbenches (*see* Simulation)
- Type II testbenches (*see* Simulation)

- Type III testbenches (*see* Simulation)
- Type IV testbenches (*see* Simulation)
- Types (*see* Data types)
- Types summary (*see* Data types)

- UFIXED (*see* Data types)
- UNAFFECTED, 124–125, 169, 172
- UNSIGNED (*see* Data types)
- unsigned* (*see* Type conversion)
- UNTIL (*see* WAIT)
- User-defined attributes, 104–106
- User-defined encoding (*see* FSM encoding)
- User-defined operators (*see* Operators)
- User-defined types, 61–70
- Using operators (*see* Operators)
- UXGA (*see* Display resolutions)

- 'VAL (*see* Attributes)
- 'VALUE (*see* Attributes)
- VARIABLE (*see* Objects)
- VCS (*see* EDA tools)
- Vending machine
 - Basic, 281
 - Glitch free, 286
- Versions (*see* VHDL versions)
- VESA, 424
- VGA
 - Circuit diagram, 426–427
 - Control signals, 428–429
 - DDC and EDID, 425–426
 - Description, 423–444
 - Display resolutions, 424, 448
 - Gamma expansion, 441
 - Image generation with file and off-chip memory, 438–441, 444
 - Image generation with file and on-chip memory, 435–438, 444
 - Image generation with hardware, 431–435, 442–443
 - Setup for the experiments, 430
 - Sync-on-green, 423
 - VGA connector, 424–425
 - VGA monitor, 423–424
- VHDL
 - 2008, 27, 80, 114, 143, 171, 218, 237, 551, 557, 567, 585, 589, 592, 595
 - Attributes (*see* Attributes)
 - Code structure, 11–24
 - Libraries (*see* Libraries)
 - Objects (*see* Objects)
 - Official documents, 3–4
 - Operators (*see* Operators)
 - Packages (*see* Packages)
 - For simulation (*see* Simulation)
 - Softwares (*see* EDA tools)
 - Syntax, 8
 - Template for state machines (*see* FSM design)
 - Types (*see* Data types)
 - Versions, 3–4
- Video interfaces (*see* VGA, DVI, FPD-Link)
- Virtex 6 (*see* FPGAs)

- WAIT
 - WAIT FOR, 159–160, 242, 253–257
 - WAIT ON, 159–160
 - WAIT statement, 151, 159–160
 - WAIT UNTIL, 159–160
- Wall clock (*see* Digital clock)
- WARNING (*see* ASSERT)
- WHEN
 - In CASE (*see* CASE statement)
 - ELSE in WHEN, 123
 - In SELECT (*see* SELECT statement)
 - WHEN statement, 121, 123–124, 125–126, 143
- WHILE (*see* LOOP)
- WIDTH, 245, 248
- WITH (*see* SELECT)
- work* library, 13
- WQUXGA (*see* Display resolutions)
- WRITE, 245, 247
- WRITELINE, 246–247
- WRITE_MODE, 245–247
- Writing to files (*see* Simulation)
- WUXGA (*see* Display resolutions)

- XC9500 (*see* CPLDs)
- XGA (*see* Display resolutions)
- Xilinx, 5, 495, 499, 500, 515
- Xilinx ISE (*see* ISE/XST)
- XNOR (*see* Logical operators)
- XOR (*see* Logical operators)