

# Index

CD information is listed by chapter and section number followed by page ranges (CD9.1:1-2). In More Depth references are listed by chapter number followed by page ranges (IMD4:5-6). Page references preceded by a single letter refer to appendices.

## A

- Absolute addresses, A13
- Abstractions, 21–22, 24
- Accumulator architectures, CD2.19:1–2
- Accumulator instructions, IMD2:7
- Acronyms, 9–10
- ACS, CD6.13:4
- Activation record, 86
- Active matrix display, 18
- Ada, 173
- add, 49–51, 301
- Adder, 292
- add immediate, 58
- add immediate unsigned, 172
- Addition, 170–176
  - carry lookahead, B38–47
  - floating point, 197–201
- Address (addressing)
  - absolute, A13
  - base, 55
  - calculation, 385, 390, 392, 402
  - exception, 342–343
  - in large-scale parallel processors, CD9.4:23–25
  - memory, 54
  - PC-relative, 98
  - physical, 511, 512, 513–514
  - translation, 512, 521–524
  - virtual, 512
- Addressing, MIPS
  - branches and jumps, 97–99, 294–295
  - decoding machine language, 100–104
  - mode summary, 100
  - 32-bit immediate operands, 95–96
- Addressing modes
  - IA-32, 138
  - MIPS, 100
  - RISC, D5–9
- add unsigned, 172
- Advanced Research Project Agency (ARPA), CD7.9:9, CD8.3:5, CD8.11:7
- Advance load, 442
- Agarwala, Tilak, CD6.13:4
- Aho, Al, CD2.19:8
- Aiken, Howard, CD1.7:3
- Air bags, 281
- Algol, CD2.19:6–7
- Aliasing, 528
- Alignment restriction, 56
- Allan, Fran, CD2.19:8
- Allocate-on-miss, 484
- Alpha architecture, CD5.12:3, D27–28
- Alto, 16, CD1.7:7–8, CD7.9:10, CD8.11:7
- ALU. *See* Arithmetic logic unit
- ALUOp, 301–305
- ALUOut, 319, 320, 327
- AMD, 136
- Amdahl, Gene, CD5.12:1
- Amdahl's law, 179, 267, 494, CD9.2:9, CD9.9:40, IMD4:5–6
- AMD Opteron, memory hierarchy, 546–550
- and (AND), 70, 301, 321, B6
- AND gate, CD3.10:5
- and immediate, 71
- Andreessen, Marc, CD8.11:7
- Antidependence, 439
- Antifuse, B77
- Antilock brakes, 281
- Apple II, CD1.7:5
- Application binary interface (ABI), 22
- Applications software, 11
- Archeological sites, 236–237
- Architectural registers, 448
- Architecture. *See* Instruction set architecture
- Arithmetic
  - addition, 170–176
  - division, 183–189
  - fallacies and pitfalls, 220–224
  - floating point, 189, 191–220
  - mean, 257–258
  - multiplication, 176–182
  - signed and unsigned numbers, 160–170
  - subtraction, 170–176
- Arithmetic-logical instructions, 292–293, 298
  - multiple-cycle implementation, 327, 329
  - single-cycle implementation, 300–318
- Arithmetic logic unit (ALU), 177, 179, 184, 187, 201
  - adders and, 292, 294
  - ALUOp, 301–305
  - ALUOut, 319, 320, 327
  - constructing, B26–38
  - control, 301–303, C4–8

- datapaths and, 286, 292, 294, 296
  - MIPS, B32–38
  - multicycle implementation, 318–340
    - 1-bit, B26–29
    - single-cycle implementation, 300–318
    - 32-bit, B29–36
  - ARM, D36–38
  - ARPANET, CD8.3:5, CD8.11:7
  - Arrays
    - of logic elements, B18–19
    - versus pointers, 130–134
  - Art, restoration of, 562–563
  - ASCII (American Standard Code for Information Interchange), 90–91
    - versus binary numbers, 162
  - Assembler directives, A5
  - Assemblers, 13, 107–108, A4, 10–17
  - Assembly language, 13, 107, A3–10
    - See also* MIPS assembly language
    - disadvantages of, A9–10
    - when to use, A7–9
  - Asserted signal, 290, B4
  - Assert signal, 290
  - Associativity, in caches, 499–502
  - Asynchronous bus, 582–583
  - Asynchronous inputs, B75–77
  - Atanasoff, John, CD1.7:3
  - AT&T Bell Labs, CD7.9:8–9
  - Atomic swap operation, CD9.3:18
  - Automatic storage class, 85
  - Availability, 573
  - Average Memory Access Time (AMAT), IMD7:1
- B**
- Bachman, Charles, CD8.11:4, 5
  - Backpatching, A13
  - Backplane, 582
  - Backus, John, CD2.19:6, 7
  - Barrier synchronization, CD9.3:15
  - Base address, 55, 100
  - Base register, 55
  - Base stations, CD8.3:9
  - Base 2 to represent numbers, 160–161
  - Basic block, 75
  - Basket, Forrest, CD7.9:9
  - Behavioral specification, B21
  - Bell Labs, CD7.9:8–9
  - Benchmarks, 254–255
    - EEMBC, 255, IMD4:17–18
    - kernel, CD4.7:2, IMD4:7–8
    - SPEC CPU, 254–255, 259–266, CD4.7:2–3, IMD4:7–8
    - SPECweb99, 262–266
    - synthetic, CD4.7:1–2, IMD4:11–12
  - Berkeley Computer Corp. (BCC), CD7.9:8, 9
  - Berkeley Software Distribution (BSD), CD7.9:9
  - Berners-Lee, Tim, CD8.11:7
  - Biased notation, 170, 194
  - Bigelow, Julian, CD1.7:3
  - Big Endian, 56, A43
  - Big-interleaved parity (RAID 3), 576–577
  - BINAC, CD1.7:4
  - Binary digits (numbers), 12, 60
    - adding and subtracting, 170–176
    - ASCII versus, 162
    - converting to decimal floating point, 196
    - converting to decimals, 164
    - hexadecimal-binary conversion table, 62
    - scientific notation, 191
    - use of, 160–161
  - Binary point, 191
  - Bit(s), 12, 60
    - in a cache, 479
    - dirty, 521
    - fields, IMD2:13–14
    - least significant, 161
    - map, 18
    - most significant, 161
    - reference/use, 519
    - sign, 163
    - sticky, 215
  - Bit error rate (BER), CD8.3:9
  - Blaauw, Gerrit, CD6.13:2
  - Block, Barbara, 156–157
  - Blocking assignment, B24
  - Block-interleaved parity (RAID 4), 577–578
  - Blocks
    - defined, 470
    - finding, 540–541
    - locating in caches, 502–504
    - placement of, 538–540
    - reducing cache misses with, 496–502
    - replacing, 504, 541–542
  - Bonding, 30
  - Boolean algebra, B6
  - Booth's algorithm, IMD3:5–9
  - Bounds check shortcut, 168
  - Branch (es)
    - addressing in, 97–99, 294–295
    - delayed, 297, 382, 418–419, A41
    - delay slot, 423
    - history table, 421
    - loop, 421–422
    - multiple-cycle implementation, 327–328, 336
    - not taken, 295, 418
    - prediction, 382, 421–423
    - prediction buffer, 421
    - taken, 295
    - target address, 294–296
    - target buffer, 423
  - Branch equal (beq), 294, 297, 300–318
  - Branch/control hazards, 379–382, 416–424
    - delayed, 297, 382, 418–419
    - dynamic branch prediction, 421–423
    - not taken, 295, 418
    - untaken, 381
    - Verilog and, CD6.7:8–9
  - Brooks, Fred, Jr., CD6.13:2
  - Bubble Sort, 129
  - Burks, Arthur W., 48, CD1.7:3, CD3.10:1
  - Buses, 291–292
    - advantages/disadvantages of, 581
    - asynchronous, 582–583
    - backplane, 582
    - basics of, 581–585
    - defined, 581, B18–19
    - master, 594
    - Pentium 4, 585–587

- processor-memory or I/O, 582
  - shared, 322–324
  - synchronous, 582–583
  - transaction, 582
  - Bypassing, 376–377
  - Byte addressing, 56
  - Byte order, A43
- C**
- C
- bit fields, IMD2:13–14
  - converting floating points to MIPS
    - assembly code, 209–213
  - development of, CD2.19:7
  - logical operations, 68–71
  - overflows, 172
  - procedures, 81–88
  - sort example, 121–129
  - strings, 92–93
  - translating hierarchy, 106–111
  - while loop in, 74–75
  - Cache coherency
    - multiprocessor, CD9.3:12–20
    - protocols, CD9.3:13, 16–18
    - snooping, CD9.3:13
    - synchronization using, CD9.3:18–20
  - Cache-coherent nonuniform memory
    - access (CC-NUMA), CD9.4:22
  - Caches
    - accessing, 476–482
    - associativity, 499–502
    - basics of, 473–491
    - bits in, 479
    - blocks, locating in, 502–504
    - blocks used to reduce misses, 496–502
    - defined, 473
    - direct-mapped, 474–475, 497
    - example of simple, 474–476
    - fully associative, 497
    - Intrinsity FastMATH processor
      - example, 485–487
    - mapping address to multiword
      - block, 480
    - memory, 20
    - memory system design to support, 487–491
    - misses, handling, 482–483, 496–502
    - multilevel, 492, 505–510
    - nonblocking, 445, 548
    - performance, measuring and improving, 492–511
    - performance with increased clock rate, 495–496
    - reducing miss penalty using multi-level, 505–509
    - set associative, 497, 504
    - split, 487
    - tags, 475, 504
    - three Cs model, 543–545
    - valid bit, 476
    - writes, handling, 483–485
  - Callee, 80, A23
  - Caller, 80, A23
  - Cal TSS, CD7.9:8
  - Capacity misses, 543
  - Carnegie Mellon University, CD6.13:5
  - Carrier signal, CD8.3:8
  - Carry lookahead, B38–47
  - Carry save adders, 181, IMD3:17–18
  - Case statement, 76
  - Cathode ray tubes (CRTs), 18
  - Cause register, 342
  - CauseWrite, 342
  - Central processor unit (CPU), 20
    - execution time, 244–245
    - performance, 245, 246–253
    - time, 244–245
  - Cerf, Vint, CD8.11:7
  - Chamberlin, Donald, CD8.11:5
  - Characters, Java, 93–95
  - Chavín de Huántar, 236–237
  - Chips, 20, 30
  - Clearing words in memory
    - arrays and, 130–132
    - comparing both methods, 133–134
    - pointers and, 132–133
  - Clock cycles, 245, B47
    - finite state machines, 332
    - multicycle implementation, 318–340
    - single-cycle implementation, 300–318
  - Clock cycles, breaking execution into
    - arithmetic-logical instruction, 327, 329
    - branches, 327–328
    - decode instruction and register fetch, 326–327
    - fetch instruction, 325–326
    - jump, 328
    - memory read, 329
    - memory reference, 327, 328
  - Clock cycles per instruction (CPI), 248–251
    - in multicycle CPU, 330–331
  - Clocking methodology, 290–292, B47
    - edge-triggered, 290–291, B47
    - level-sensitive, B74–75
    - timing methodologies, B72–77
  - Clock period, 245, B47
  - Clock rate, 245
  - Clocks, B47–49
  - Clock skew, B73–74
  - CLU, CD2.19:7
  - Clusters, CD9.1:4, CD9.5:25–26
  - CMOS (Complementary Metal Oxide Semiconductor), 31, 264
  - Coarse-grained multithreading, CD9.7:31–33
  - Cobol, CD2.19:6, CD8.11:4
  - Cocke, John, CD2.19:8, CD6.13:2, 4
  - Codd, Ted, CD8.11:4, 5
  - Code generation, CD2.12:9
  - Code motion, 119
  - Code size, fallacy of using, IMD4:18–19
  - Coherence problem, 595
  - Cold-start misses, 543
  - Collision misses, 543
  - Color, 292
  - Colossus, CD1.7:3
  - Combinational control units, C4–8
  - Combinational elements, 289
  - Combinational logic, B5, 8–20, 23–25
  - Compact disks (CDs), 25

- Compaq Computers, CD8.11:6
  - Comments, 50
  - Commit unit, 443
  - Common subexpression elimination, 117
  - Compilers
    - C, 107
    - functions of, 11–12
    - historical development of, CD2.19:7–8
    - how they work, CD2.12:1–9
    - Java, 114–115
    - optimization, 116–121
    - structure of, 116
    - translating high-level language into instructions that hardware can execute, 12–15, A5–6
  - Compulsory misses, 543
  - Computers
    - applications, 5–7
    - components of, 15–16
    - historical development of, CD1.7:1–10
    - organization of, 16
    - what it looks like inside, 18–22
  - Computer technology, advances in, 4
  - Conditional branches, 72–73
  - Condition codes, 140
  - Conflict misses, 543
  - Constant folding, 118
  - Constant propagation, 118
  - Constants, 57, 58
    - loading 32-bit, 96
  - Constellations, CD9.5:26
  - Context switch, 530
  - Control, 20
    - hardwired, 348, CD5.12:2
    - pipelined, 399–402
  - Control Data Corp. (CDC), CD1.7:5, CD6.13:2
  - Control hazards. *See* Branch hazards
  - Controller time, 570
  - Control signals
    - list of, 306, 324
    - write, 290, 294
  - Control unit
    - adding, 299
    - combinational, C4–8
    - designing main, 303–312
    - exceptions, 340–346
    - fallacies and pitfalls, 350–352
    - finite state machines, 330, 331–340, C8–20
    - interrupts, 340–341
    - jumps, 313–314
    - microprogramming, 330, CD5.7:4–10
    - multicycle implementation, 318–340
    - single-cycle implementation, 300–318
  - Coonen, Jerome T., CD3.10:7
  - Copy back, 521
  - Copy propagation, 118
  - Corbato, John, CD7.9:7, 11
  - Correlating predictors, 423
  - Cosmic Cube, CD9.11:52
  - C++, CD2.19:7
  - CPU. *See* Central processor unit
  - Cray, Seymour, CD1.7:5, CD3.10:4, CD6.13:2
  - Cray Research, Inc., CD1.7:5, CD3.10:4–5, CD6.13:5
  - Critical word first, 482
  - Crossbar network, CD9.6:30
  - CTSS (Compatible Time-Sharing System), CD7.9:7–11
  - Culler, David, 157
  - Culler Scientific, CD6.13:4
  - Cutler, David, CD7.9:9
  - Cydrome Co., CD6.13:4, 5
  - Cylinder, use of term, 569
- D**
- Dahl, Ole-Johan, CD2.19:7
  - Databases, history of, CD8.11:4–5
  - Data General, CD8.11:6
  - Data hazards
    - defined, 376–379
    - forwarding, 402–412
    - load-use, 377
    - stalls, 413–416
  - Data parallelism, CD9.11:48
  - Datapath, 20
    - building a, 292–300
    - elements, 292
    - fallacies and pitfalls, 350–352
    - jumps, 313–314
    - logic design conventions, 289–292
    - multicycle implementation, 318–340
    - operation of, 306–312
    - pipelined, 384–399
    - single-cycle implementation, 300–318
  - Data rate, 598
  - Data segment, A13, 20
  - Data selector, 286
  - Data transfer instructions, 54–55
  - Data types, Verilog, B21–22
  - Dawson, Todd, 157
  - Dead code elimination, 118
  - Dead store elimination, 118
  - Deasserted signal, 290, B4
  - Deassert signal, 290
  - Debugging information, A13
  - DEC (Digital Equipment Corp.), CD1.7:5, CD4.7:2, CD7.9:9, CD8.11:11
  - Decimal numbers, 60, 161
    - converting binary numbers to, 164
    - converting binary to decimal floating point, 196
    - dividing, 183
    - multiplying, 176–177
    - scientific notation, 189
  - Decision-making instructions, 72–74
  - Decoders, B8–9
  - Decoding, 333
  - Dedicated register, CD2.19:2
  - Defects, 30
  - Delayed branch, 297, 382, 418–419, A41
  - Dell Computer Corp.
    - SPEC CPU benchmarks, 254–255, 259–266
    - SPECweb99 benchmark, 262–266

- DeMorgan's laws, B6
  - DeMorgan's theorems, B11
  - Dependability, disk, 569–580
  - Dependence detection, 406
  - Dependent instructions, 403
  - Desktop computers, 5
    - performance benchmarks, 255
  - Destination register, 64
  - Deutsch, Peter, CD7.9:8
  - D flip-flop, B51–53
  - Dhrystone synthetic benchmark,
    - CD4.7:2, IMD4:11–12
  - Dies, 30
  - Digital cameras, 236–237, 603–606
  - Digital signal-processing extensions, D19
  - DIMMs. *See* Dual inline memory modules
  - Directives, data layout, A14–15
  - Direct-mapped cache, 474–475, 497
  - Direct memory access (DMA), 594–596
  - Directories, CD9.4:24
  - Dirty bit, 521
  - Disabled people, technology for, 366–367
  - Disk(s)
    - arrays, IMD8:2
    - controller, 570, 571
    - drives, 19, 20
    - fallacies and pitfalls, 606–609
    - read time, 570–571
    - storage and dependability, 569–580,
      - CD8.11:1–4
  - Dispatch, 350
  - Displacement addressing, 100
  - Distributed block-interleaved parity
    - (RAID 5), 578
  - Distributed memory, CD9.4:22
  - Distributed shared memory (DSM),
    - CD9.4:24
  - divide, 188–189
  - Dividend, 183
  - divide unsigned, 188–189
  - Division, 183–189
  - Divisor, 183
  - Don't-care terms, 303, B16–18
  - Double, 192
  - Double extended precision, 218
  - Double precision, 192
  - Double data rate synchronous DRAMs
    - (DDD SDRAMs), 490–491
  - DRAM. *See* Dynamic random access memory
  - Dual inline memory modules
    - (DIMMs), 22
  - DVD drive, 19, 20
  - DVDs (digital video disks), 25
  - Dynamically linked libraries (DLLs),
    - 112–114
  - Dynamic branch prediction, 382,
    - 421–423
  - Dynamic data, A22
  - Dynamic multiple issue, 433, 442–445
  - Dynamic pipeline scheduling, 443–445
  - Dynamic random access memory
    - (DRAM), 20, 469, 487–488,
      - 490–491, 513, B60, 63–65
    - historical development of, CD7.9:3–4
- E**
- Early restart, 481–482
  - Eckert, J. Presper, CD1.7:1, 2, 4, CD7.9:1
  - Eckert-Mauchly Computer Corp., CD1.7:4
  - Edge-triggered clocking methodology,
    - 290–291, B47
  - EDSAC (Electronic Delay Storage Automatic Calculator), CD1.7:2,
    - CD5.12:1
  - EDVAC (Electronic Discrete Variable Automatic Computer), CD1.7:1–2
  - EEMBC benchmarks, 255, IMD4:17–18
  - 802.11 standard, CD8.3:9–10
  - Eispack, CD3.10:3
  - Elaborations, 8
  - Elapsed time, 244
  - Ellison, Larry, CD8.11:5
  - Embedded computers, 6–8, CD1.7:8–9, A7
    - performance benchmarks, 255,
      - IMD4:17–18
  - EMC, CD8.11:6
  - Emulation, CD5.12:1–2
  - Encoder, B9
  - Energy efficiency problems, 263–265
  - Engelbart, Doug, 16
  - ENIAC (Electronic Numerical Integrator and Calculator), CD1.7:1–2, 3,
    - CD7.9:1
  - Environmental problems, technology
    - and, 156–157
  - EPCWrite, 342
  - Error-correcting codes, B65
  - Error-detecting codes, B65–67
  - Ethernet, 26, CD8.3:5, CD8.11:7–8,
    - IMD8:1–2
  - Evolution versus revolution,
    - CD9.10:45–47
  - Exception enable, 532
  - Exception program counter (EPC), 173,
    - 341–342, 429–431
  - Exceptions, 173, A33–38
    - address, 342–343
    - control checking of, 343–346
    - defined, 340–341
    - handling of, 341–343, A35, 36–38
    - imprecise, 432
    - pipeline, 427–432
  - Executable file, 109
  - Execution time, 242, 244–245
    - use of total, 257–259
  - Executive process, 529
  - Exponent, 191
  - Extended accumulator, CD2.19:2
  - External labels, A11
- F**
- Failures
    - mean time between failures
      - (MTBF), 573
    - mean time to failure (MTTF), 573, 574
    - mean time to repair (MTTR), 573, 574
    - reasons for, 574
    - synchronizer, B76
  - Fallacies, 33
  - False sharing, CD9.3:14
  - Fanout, 32
  - Fast carry, B38–47

- Fetch-on-miss/write, 484
  - Field programmable devices (FPDs), B77–78
  - Field programmable gate arrays (FPGAs), B77
  - Fields
    - defined, 61
    - MIPS, 63–64
  - File system benchmarks, 598–599
  - Fine-grained multithreading, CD9.7:31–33
  - Finite state machines, 330, 331–340, B47–72, C8–20
  - Firewire, 582, 583
  - Firmware, CD5.12:2
  - Fisher, Josh, CD6.13:4
  - Fishman, Harvey, 366–367
  - Flags, 140
  - FLASH, 23, 25
  - Flat-panel display, 18
  - Flip-flops, 290, B50–53
  - Floating point, 189, 191–220
    - addition, 197–201
    - converting binary to decimal floating point, 196
    - defined, 191
    - historical development of, CD3.10:1–9
    - IA-32, 217–220
    - MIPS, 206–213
    - multiplication, 202–205
    - representation, 191–197
    - rounding, 214–215
  - Floating Point Systems, CD6.13:4, 5
  - Floating vectors, CD3.10:2
  - Floppy disks, 25, CD1.7:6
  - Floppy drives, 25
  - Flush instructions, 418
  - Flynn, Michael, CD6.13:3
  - Formal parameter, A16
  - Forrester, J., CD7.9:1
  - FORTRAN, CD2.19:6, 7–8
    - overflows, 172, 173
  - Forwarding, 376–377, 402–412, CD6.7:3
  - Forward reference, A11
  - Fraction, 191, 193
  - Frame buffer, 18
  - Frame pointer, 86
  - Front end of compiles, CD2.12:1–9
  - Fully associative cache, 497
  - Fully connected network, CD9.6:28
  - Function code, 63
- G**
- Gates, B7–8, C4–8
  - Gateways, CD8.3:6
  - General-purpose register (GPR), 135, 138, CD2.19:2–3
  - Generate, carry lookahead, B39–47
  - Geometric mean, IMD4:9–11
  - Gibson, Garth, CD8.11:6
  - Global common subexpression elimination, 118
  - Global labels, A11
  - Global miss rate, 509
  - Global optimization, 117–121, CD2.12:4–6
  - Global pointer, 85
  - Goldstine, Herman H., 48, CD1.7:1–2, 3, CD3.10:1
  - Google, CD9.8:34–39
    - News, 465
  - Gosling, James, CD2.19:7
  - Graph coloring, CD2.12:7–8
  - Graphics display, 18
  - Gray, Jim, CD8.11:5
  - Gray-scale display, 18
  - Guard, 214–215
- H**
- Half words, 94
  - Hamming code, B67
  - Handler, 533
  - Handshaking protocol, 583–584
  - Hard disk, magnetic, 23
  - Hard drive, 19, 20
  - Hardware
    - description language, B20–25
    - functions of, 15
    - performance affected by, 10
    - synthesis tools, B21
  - Hardwired control, 348, CD5.12:2
  - Harvard architecture, CD1.7:3
  - Hazards
    - See also* Pipelining hazards
    - detection unit, 413–415
  - Heap, allocating space for data on, 87–88
  - Heat sink, 22
  - Held, Gerald, CD8.11:5
  - Hewlett-Packard, CD2.19:5, CD3.10:6–7, CD4.7:2
    - PA-RISC 2.0, D34–36
  - Hexadecimal-binary conversion table, 62
  - Hi, 181
  - High-level optimization, 116–117
  - High-level programming languages
    - advantages of, 14–15
    - architectures, CD2.19:4
    - defined, 13
    - translating into instructions that hardware can execute, 12–15
  - Hit(s)
    - Average Memory Access Time (AMAT), IMD7:1
    - defined, 470
    - rate/ratio, 470–471
    - time, 471
  - Hitachi, SuperH, D39–40
  - Hold time, B53
  - Hot swapping, 579
  - Hubs, CD8.3:7
- I**
- IBM
    - disk storage, CD8.11:1–4
    - early computers, CD1.7:5
    - floating points, CD3.10:2, 3–4
    - floppy disks, CD1.7:6, CD8.11:2
    - history of programming languages, CD2.19:6
    - microprogramming, CD5.12:1–2
    - multiple issue, CD6.13:4

- PowrPC, D32–33, IMD2:17–20, IMD3:10
- RAID, CD8.11:6
- Stretch computer, CD6.13:1–2
- virtual memory, CD7.9:5–7, 10
- Winchester disks, CD8.11:2, 4
- IEEE 754 floatation-point standard, 193–196, CD3.10:7–9
- If-then-else statements, compiling into
  - conditional branches, 72–73
- Immediate addressing, 100
- Implementation, 22, 24
- Imprecise interrupts/exceptions, 432, CD6.13:3
- IMS, CD8.11:4
- Induction variable elimination, 119–120
- Infinity, 193
- Ingres, CD8.11:5
- In-order commit, 445
- In-order completion, 445
- Input devices, 15, 566, A38–40
- Input don't cares, B16
- Input operation, 582
- Inputs, asynchronous, B75–77
- Instruction decode, 385, 390, 392, 402
- Instruction encoding, MIPS floating-point, 208
- Instruction group, 440
- Instruction fetch, 385, 388–389, 392, 400
- Instruction format, 61
- Instruction latency, 452
- Instruction-level parallelism (ILP), CD9.7:33, 433, CD6.13:5
- Instruction mix, 253
- Instruction register (IR), 319, 321
- Instruction sets
  - addressing, 95–105
  - architecture, 22, 24
  - compiler optimization, 116–121
  - decision-making instructions, 72–74
  - defined, 48
  - designing, for pipelining, 374–375
  - historical development of, CD2.19:1–9
  - logical operations, 68–71
  - operands of hardware, 52–60
  - operations of hardware, 49–52
  - to process text, 90–95
  - representing instructions to computer, 60–68
  - styles, IMD2:7–9
  - supporting procedures, 79–90
  - translating and starting a program, 106–115
- Integers, signed versus unsigned, 165
- Integrated circuits (ICs)
  - costs, IMD1:1–2
  - defined, 20, 27–28
  - how they are manufactured, 28–33
- Integrated Data Store (IDS), CD8.11:4
- Intel, CD1.7:5, 6, CD8.11:8
  - See also* Pentium 4
  - 8086, 135, CD2.19:2, 4, 5
  - 8087, 135, CD3.10:7
  - 80286, 135, CD2.19:5
  - 80386, CD2.19:5
  - 80486, 135, CD2.19:5
  - iSC 860 and Paragon, CD9.11:52
  - Pentium and Pentium Pro, 135, CD2.19:5, 448–450
  - SPEC CPU benchmarks, 254–255, 259–266
  - SPECweb99 benchmark, 262–266
- Intel IA-32, 59
  - addressing modes, 138
  - complexity of, 347–348
  - conclusions, 142–143
  - fallacies and pitfalls, 143–144
  - floating point, 217–220
  - historical development of, 134–137, CD2.19:4–5
  - instruction encoding, 140–142
  - integer operations, 138–140
  - registers, 137–138
- Intel IA-64, 435, CD5.12:3
  - architecture, 440–442, CD6.13:4–5
- Intel Streaming SIMD Extensions (SSE), 135–136
- Intel Streaming SIMD Extension 2 (SSE2), 136
- floating points, 220
- Interface message processor (IMP), CD8.3:5
- Interference graph, CD2.12:7
- Interleaving, 489
- Intermediate representation, CD2.12:2–3
- Internet, CD8.11:7
  - news services, 464–465
- Internetworking, CD8.3:1–4
- Interrupt-driven I/O, 590–591
- Interrupts, 173, A33–38
  - handler, A33
  - imprecise, 432, CD6.13:3
  - priority levels, 591–593
  - use of term, 340–341
- Intrinsity FastMATH processor example, 485–487, 524
- Invalid operations, 193
- I/O
  - buses, 582
  - communicating with processor, 590–591
  - designing a system, 600–603
  - devices, 15, 566, A38–40
  - digital camera example, 603–606
  - diversity of, 568
  - fallacies and pitfalls, 606–609
  - giving commands to devices, 589–590
  - historical development of, CD8.11:1–9
  - instructions, 590
  - interfacing devices to processor, memory, and operating system, 588–596
  - interrupt-driven, 590–591
  - interrupt priority levels, 591–593
  - measuring performance, 567
  - memory-mapped, 589–590
  - performance, 597–600
  - rate, 598
  - requests, 568
  - transferring data between devices and memory, 593–595
- Issue packet, 435
- Issue slots, 434

**J**

Java  
 bytecode, 114, CD2.14:1, 2  
 characters and strings, 93–95  
 compiling, CD2.14:4–6  
 development of, CD2.19:7  
 interpreting, CD2.14:1–3  
 invoking methods, CD2.14:6  
 logical operations, 68–71  
 translating hierarchy, 114–115  
 sort and swap, CD2.14:6–13  
 while loop, CD2.14:3–4, 5–6  
 Java Virtual Machine (JVM), 115,  
 CD2.14:3  
 Jhai Foundation, PC network, 44–45  
 Jobs, Steven, CD1.7:5  
 Johnson, Reynold B., CD8.11:1  
 Joy, Bill, CD7.9:9  
 J-type, 97  
 jump, 73, 77, 80, 89, 296  
   addressing in, 97–99  
   datapath and control and, 313–314,  
   321, 328, 336  
 Jump address table, 76, 77,  
 IMD2:15–16  
 jump-and-link, 79–80, 89  
 jump register, 76  
 Just-in-Time (JIT) compiler, 115

**K**

Kahan, William, CD3.10:5–7, 8, 9  
 Kahn, Robert, CD8.11:7  
 Karnaug maps, B18  
 Katz, Randy, CD8.11:6  
 Kay, Alan, CD2.19:7  
 Kernel benchmarks, CD4.7:2, IMD4:7–8  
 Kernel process, 529  
 Knuth, Donald, CD2.19:8

**L**

Labels, external/global and local, A11  
 Lampson, Butler, CD7.9:8, 11

Laptop computers, performance versus  
 power versus energy efficiency,  
 263–265  
 Latches, B59–53  
 Latency  
   instruction, 452  
   pipeline, 383  
 Leaf procedures, 83, 93  
 Least recently used (LRU), 504,  
 518, 519  
 Least significant bit, 161  
 Level-sensitive clocking, B74–75  
 Link editor, 109  
 Linkers, 108–111, A4, 18–19  
 Linpack, CD3.10:3, CD4.7:2  
 Linux, 11, CD7.9:11  
 Liquid crystal displays (LCDs), 18  
 Lisp, CD2.19:6  
 Little Endian, 56, A43  
 Live range, CD2.12:7  
 Livermore Loops, CD4.7:2  
 Lo, 181  
 Load, 54, 57  
   advanced, 442  
   byte, 91, 164  
   byte unsigned, 164  
   half, 94, 164  
   halfword unsigned, 164  
   linked, CD9.3:19  
   locked, CD9.3:19  
   upper immediate, 95  
   word, 54, 57, 59, 294, 300–318  
 Loader, 112  
 Loading, A19–20  
 Loading 32-bit constant, 96  
 Load-use data hazard, 377  
 Local area networks (LANs), 26,  
 CD8.3:5–8, CD8.11:7–8  
 Locality, principle of, 468–469  
 Local labels, A11  
 Local miss rate, 509  
 Local optimization, 117–121,  
 CD2.12:3–4  
 Lock, CD9.1:5  
 Lock variables, CD9.3:18

**Logic**

arrays of logic elements, B18–19  
 combinational, B5, 8–20, 23–25  
 equations, B6–7, C12–13  
 sequential, B5, 55–57  
 two-level, B10–14  
 Logical operations, 68–71, B6,  
 IMD2:21–22  
 Logic design conventions, 289–292  
 Long-haul networks, CD8.3:5  
 Long instruction word (LIW),  
 CD6.13:4  
 Lookup tables (LUTs), B78  
 Loops, 74–75  
   branch, 421–422  
   unrolling, 117, 438–440  
 Lorie, Raymond, CD8.11:5

**M**

Machine code, 61  
 Machine language, 61, A3  
   decoding, 100–104  
   MIPS floating-point, 207  
   object file and, 108  
 MacOS, 11  
 MacOS, A4, 15–17  
 Magnetic disks, 23, 569  
   differences between main memory  
   and, 24  
   memory hierarchies and, 469, 513  
 Magnetic resonance imaging (MRI),  
 622–623  
 Magnetic tape, 25  
 Main memory, 23  
   differences between magnetic disks  
   and, 24  
 Make the common case fast, 267, 285  
 Mark machines, CD1.7:3  
 Mask, 70  
 Mauchly, John, CD1.7:1, 2, 4  
 McCarthy, John, CD2.19:6, CD7.9:7, 11  
 McKeeman, William, CD2.19:8  
 Mealy, George, 338  
 Mealy machine, 338, 340, B68



- Mean time between failures (MTBF), 573
- Mean time to failure (MTTF), 573, 574, 606
- Mean time to repair (MTTR), 573, 574
- Megabyte, 23
- Memories, 290
- Memory, 8
  - access, 385, 390, 392, 402
  - allocation, 87–88
  - Average Memory Access Time (AMAT), IMD7:1
  - board, 20
  - cache, 20
  - cards, 25
  - consistency model, CD9.3:15
  - defined, 20, 23
  - direct memory access (DMA), 594–596
  - distributed, CD9.4:22, 24
  - dynamic random access (DRAM), 20, 469, 487–488, 490–491, 513, B60, 63–65
  - historical development of, CD7.9:1–12
  - main, 23
  - mapping, 512
  - nonvolatile, 23
  - operands, 54–55
  - primary, 23
  - random access (RAM), 20
  - read only (ROM), B14, 16, C13–19
  - secondary, 23
  - shared, CD9.1:4–5, CD9.4:22, 24
  - static random access (SRAM), 20, 469, B57–60
  - transferring data between devices and, 593–595
  - unit, 292
  - usage, A20–22
  - virtual, 511–538
  - volatile, 23
- Memory data register (MDR), 319, 328
- Memory elements
  - latches, flip-flops, and register files, B49–57
  - SRAMs and DRAMs, B57–67
- Memory hierarchy
  - caches, 473–511
  - defined, 469
  - fallacies and pitfalls, 550–552
  - framework for, 538–545
  - historical development of, CD7.9:5–7
  - levels, 470–471
  - methods for building, 469–470
  - overall operation of, 527–528
  - Pentium P4 and AMD Opteron, 546–550
  - trends for, 553–555
  - virtual, 511–538
- Memory-mapped I/O, 589–590
- Memory-memory instructions, IMD2:8
- Memory reference, 327, 328, 334–335
- MESI cache coherency protocol, CD9.3:16, 18
- Message passing, CD9.1:6, CD9.4:22–23
- Metastability, B75–76
- MFLOPS (million floating-point operations per second), IMD4:15–17
- Microarchitecture, 448
- Microcode, 348
- Microinstructions, 348–349, CD5.7:1
  - fields, CD5.7:3, 5–9
  - format, CD5.7:2–4
- Microoperations, 348
- Microprocessors
  - first, CD1.7:5
  - future of, CD9.10:44–45
- Microprogramming
  - controller, 348, CD5.12:2
  - creating a program, CD5.7:4–10
  - defined, 330, 346
  - fallacies and pitfalls, 350–352
  - historical development of, CD5.12:1–4
  - implementing the program, CD5.7:10–12
  - microinstruction format defined, CD5.7:2–4
  - simplifying design with, CD5.7:1–13
- Microsoft Corp., CD1.7:5, CD7.9:10, CD8.11:5, 6
- Minicomputers, first, CD1.7:5
- Minterms, B12
- MIPS, 49
  - addressing, 95–105
  - allocation of memory, 87
  - arithmetic logic unit (ALU), B32–38
  - compiling statements into, 50–51
  - decision-making instructions, 72–73
  - exception code, 535
  - fields, 63–64
  - floating point, 206–213
  - implementation, 285–289
  - instruction encoding table, 64, 103
  - instruction set, 49
  - logical operations, 68–71
  - machine language, summary of, 67, 78, 90
  - mapping registers into numbers, 60–68
  - operands, summary of, 59, 67, 71, 89, 105, 169
  - registers, 52–53, 79–80, 85, 88, 532
  - RISC core subset, D9–16, 20–24
  - RISC instructions for MIPS16, D41–43
  - RISC instructions for MIPS64, D25–27
  - translating assembly into machine language, 65–66
- MIPS assembly language
  - add, 49–51
  - add immediate, 58, 59
  - add immediate unsigned, 172
  - add unsigned, 172
  - AND, 69, 70
  - and immediate, 71, 89
  - conditional and unconditional branches, 72–73
  - divide, 188–189
  - divide unsigned, 188–189
  - floating point, 207
  - jump, 73, 80
  - jump address table, 76
  - jump-and-link, 79–80
  - load word, 54–59, 294
  - move from hi, 181
  - move from lo, 181
  - multiply, 181

- multiply unsigned, 181
  - nor (NOR), 69, 70
  - or (OR), 69, 70
  - or immediate, 71, 89
  - set on less than, 75
  - set on less than immediate, 77, 165
  - set on less than immediate
    - unsigned, 165
  - set on less than unsigned, 165
  - shifts, 69
  - store word, 54–59, 294
  - subtract, 49–51
  - subtract unsigned, 172
  - summary of, 51, 59, 67, 71, 77, 89, 105, 169, 175, 190, 207, 226–228
  - xor, IMD2:21–22
  - MIPS assembly language, R2000
    - addressing modes, A45–47
    - arithmetic and logical instructions, A51–57
    - assembler syntax, A47–49
    - branch instructions, A59–63
    - comparison instructions, A57–59
    - constant manipulating instructions, A57
    - data movement instructions, A70–73
    - encoding instructions, A49
    - exception and interrupt instructions, A80–81
    - floating-point instructions, A73–80
    - instruction format, A49–51
    - jump instructions, A63–64
    - load instructions, A66–68
    - store instructions, A68–70
    - trap instructions, A64–66
  - MIPS (million instructions per second)
    - equation, 268
    - peak versus relative, IMD4:13–14
    - problem with using as a performance measure, 268–270
  - Mirroring, 575
  - Miss, 470
  - Misses
    - Average Memory Access Time (AMAT), IMD7:1
    - cache, 482–483, 496–502
    - capacity, 543
    - cold-start, 543
    - collision, 543
    - compulsory, 543
    - conflict, 543
    - TBL, 531
  - Miss penalty, 471
    - reducing, using multilevel caches, 505–509
  - Miss rate/ratio, 471
    - global, 509
    - local, 509
  - Mitsubishi, M32R, D40–41
  - Moore, Edward, 338
  - Moore, Gordon, 28
  - Moore machine, 338, B68
  - Moore’s law, 28, 181
  - Mosaic, CD8.11:7
  - Most significant bit, 161
  - Motherboard, 19, 20
  - Motorola
    - PowrPC, D32–33, IMD2:17–20, IMD3:10
    - 68881, CD3.10:8
  - Mouse, 16–17
  - move from hi, 181
  - move from lo, 181
  - Move from system control, 173
  - M32R, D40–41
  - Multicomputers, CD9.11:52
  - MULTICS (Multiplexed Information and Computing Service), CD7.9:8
  - Multicycle implementation, 318–340
  - Multiflow Co., CD6.13:4
  - Multilevel caching, 492, 505–510
  - Multimedia extensions of desk-top/server RISCs, D16–19
  - Multiple instruction multiple data (MIMD), CD9.11:51–53
  - Multiple instruction single data (MISD), CD9.11:51
  - Multiple issue
    - defined, 433
    - dynamic, 433, 442–445
    - IBM’s work on, CD6.13:4
    - static, 433, 435–442
  - Multiplexors, 286, B9–10
  - Multiplicand, 176
  - Multiplication, 176–182
    - floating point, 202–205
  - Multiplier, 176
  - multiply, 181
  - multiply unsigned, 181
  - Multiprocessors
    - connected by a network, CD9.4:20–25
    - connected by a single bus, CD9.3:11–20
    - defined, CD9.1:4, CD9.11:52
    - future of, CD9.10:43–44
    - history development of, CD9.11:47–55
    - inside a chip and multithreading, CD9.7:30–34
    - networks, CD9.4:20–25, CD9.6:27–30
    - programming, CD9.2:8–10
    - types of, CD9.1:4–8
  - Multistage network, CD9.6:29–30
  - Multithreading, CD9.7:30–34
- ## N
- Name dependence, 439
  - NaN (Not a Number), 193
  - NAND gate, B8
  - NCR, CD8.11:6
  - Negation shortcut, 166
  - Nested procedures, 83–85
  - Netscape, CD8.11:7
  - Network bandwidth
    - defined, CD9.6:27
    - fully connected, CD9.6:28
    - total, CD9.6:27–28
  - Networks, 25–27
    - characteristics of, CD8.3:1
    - crossbar, CD9.6:30
    - internetworking, CD8.3:1–4
    - local area, CD8.3:5–8

- long-haul, CD8.3:5
  - multiprocessors connected by,
    - CD9.4:20–25, CD9.6:27–30
  - multistaged, CD9.6:29–30
  - Pentium 4, 585–587
  - wireless local area, CD8.3:8–10
  - Next-state function, 331, B67, C12–13, 21–27
  - No-allocate-on-write, 484
  - No-fetch-on-write, 484
  - Nonblocking assignment, B24
  - Nonblocking caches, 445, 548
  - Nonuniform memory access (NUMA)
    - multiprocessors, CD9.1:6, CD9.4:22
  - Nonvolatile memory, 23
  - Nonvolatile storage device, 569
  - Nop, 413–414
  - nor (NOR), 70, 301, B8
  - Normalized number, 189
  - Northrop, CD1.7:4
  - NOT, 70, B6
  - Numbers
    - ASCII versus binary, 162
    - base to represent, 160–161
    - converting binary to decimal, 164
    - loads, 164
    - negative and positive, 165
    - shortcuts, 166–168
    - sign and magnitude, 162
    - sign bit, 163
    - signed and unsigned, 160–170
    - two's complement representation, 163
  - Nygaard, Kristen, CD2.19:7
- O**
- Oak, CD2.19:7
  - Object files
    - defined, 108, A10
    - format, A13–14
    - linking, 109–111
  - Object-oriented language
    - defined, 130, CD2.14:1
    - Javas, CD2.14:1–13
  - Offset, 55, 56
  - Opcode, 63, 303, 305, 306
  - Open Source Foundation, CD7.9:9
  - Open Systems Interconnect (OSI)
    - CD8.3:2
  - Operands
    - for computer hardware, 52–60
    - constant or immediate, 57
    - memory, 54–55
    - MIPS, summary of, 59, 67, 71, 89, 105, 169
    - MIPS floating point, 207
  - Operating systems
    - examples of, 11
    - functions of, 11–12, 588–589
    - historical development of, CD7.9:7–11
  - Operations, for computer hardware, 49–52
  - Operators, Verilog, B21–22
  - Optical disks, 25
  - Optimizations
    - high-level, 116–117
    - local and global, 117–120, CD2.12:3–6
    - summary of, 120–121
  - or (OR), 70, 301, 321, B6
  - Oracle, CD8.11:5
  - or immediate, 71
  - Out-of-order execution, 445
  - Output devices, 15, A38–40
  - Output don't cares, B16
  - Output operation, 582
  - Overflow, CD3.10:5
    - adding and subtracting and, 171–174
    - division and, 189
    - exceptions, detection of, 343
    - floating point and, 192
    - multiplying and, 181
  - Overlays, 511–512
- P**
- Packets, CD8.3:5
  - Page, 512
    - placing and find, 515–516
  - Page faults, 512, 514, 516–521, 531
  - Page offset, 513, 514
  - Page table, 515–516
  - Palmer, John F., CD3.10:7
  - Parallel processing program, CD9.1:4, CD9.2:8–10, CD9.4:22–23
    - addressing, CD9.4:23–25
    - fallacies and pitfalls, CD9.9:39–42
  - PA-RISC 2.0, D34–36
  - Parity
    - big-interleaved (RAID 3), 576–577
    - block-interleaved (RAID 4), 577–578
    - distributed block-interleaved (RAID 5), 578
  - Parsing, CD2.12:1
  - Pascal, CD2.19:6–7
  - Patterson, David, CD8.11:6
  - PC-relative addressing, 98, 100
  - PCSpim, A42, CDA:1–3
  - PCSrc control and signal, 305
  - PCWrite, 321
  - PCWriteCond, 321
  - Peer-to-peer architecture, CD8.3:9–10
  - Pentium 4
    - buses and networks of, 585–587
    - implementation of, 348–350
    - manufacturing of, 28–33
    - memory hierarchies, 546–550
    - pipeline, 448–450
  - Pentium processors
    - SPEC CPU benchmarks, 254–255, 259–266
    - SPECweb99 benchmark, 262–266
  - Performance
    - See also* Pipelining
    - benchmarks, 254–255, IMD4:7–8, 11–18
    - of caches, 253
    - of caches, measuring and improving, 492–511
    - comparing, 252–253, 256–259, 425–426
    - CPU, 245, 246–253
    - defined, 241–244
    - equation, 248–249
    - evaluating, 254–259
    - factors affecting, 251

- fallacies and pitfalls, 266–270
  - historical review, CD4.7:1–4
  - how hardware and software affect, 10
  - I/O, 597–600
  - measuring, 244–246
  - per unit cost of technologies, 27
  - of the pipeline, 253
  - relative, 243–244
  - reports, 255–256
  - single-cycle machines and, 315–318
  - SPEC CPU benchmarks, 254–255, 259–266, CD4.7:2–3, IMD4:7–8
  - SPECweb99 benchmark, 262–266
  - system, 245
  - versus power and energy efficiency, 263–265
  - Personal computers, early, CD1.7:5–8
  - Peterman, Mark, 366–367
  - Physical addresses, 511, 512, 513–514
  - Physically addressed cache, 528
  - Physical page number, 513
  - Pipeline stalls, 377–379, 413–416, CD6.7:5–7
  - Pipelining
    - advanced methods for extracting more performance, 432–445
    - control, 399–402
    - datapath, 384–399
    - defined, 370
    - designing instruction sets for, 374–375
    - exceptions, 427–432
    - fallacies and pitfalls, 451–384
    - forwarding, 376–377, 402–412, CD6.7:3
    - graphic representation, 395–399
    - historical development of, CD6.13:1–13
    - instruction execution sped up by, 372–374
    - latency, 383
    - overview of, 370
    - Pentium 4 example, 448
    - stalls, 377–379, 413–416, CD6.7:5–7
    - Verilog used to describe and model, CD6.7:1–9
  - Pipelining hazards, branch/control, 379–382, 416–424, CD6.7:8–9
  - Pipelining hazards data
    - defined, 376–379
    - forwarding, 402–412
    - load-use, 377
    - stalls, 413–416
    - structural, 375
  - Pitfalls, 33–34
  - Pixels, 18
  - Pointers, arrays versus, 130–134
  - Poison, 442
  - Polling, 590
  - Pop, 80
  - Pop-up satellite archival tags (PSATs), 156–157
  - Positive numbers, multiplying, 176–180
  - Power, 30–32
    - consumption, problems with, 263–265
  - PowerPC
    - addressing, IMD2:17–20
    - instructions, D32–33
    - multiply-add instruction, IMD3:10
  - Prediction, 382, 421–423
    - IA-64, 441
  - Primary memory, 23
  - Procedure call conventions, A22–33
  - Procedure call frame, A23
  - Procedures
    - allocating space for data on heap, 87–88
    - allocating space for data on stack, 86
    - C, 81–88
    - defined, 79
    - frame, 86
    - inlining, 116
    - leaf, 83, 93
    - nested, 83–85
    - preserved versus not preserved, 85
    - recursive, A26, 29
    - steps, 70
  - Processor, 20
    - communicating with, 590–591
    - cores, 6–7
    - memory buses, 582
  - Process switch, 530
  - Product of sums, B10–12
  - Product terms, B12
  - Program counter (PC), 80, 292
  - Programmable logic arrays (PLAs), B12–14, C7, 19–20
  - Programmable logic devices (PLDs), B77
  - Programmable read only memory (PROM), B14, 16
  - Programming languages, history of, CD2.19:6–7
  - Propagate, carry lookahead, B39–47
  - Propagation time, B77
  - Protection group, 576
  - Protocol families/suites, CD8.3:1–2
  - Protocol stack, CD8.3:3
  - Pseudodirect addressing, 100
  - Pseudoinstructions, 107, A17
  - Push, 80
  - Putzolu, Gianfranco, CD8.11:5
- Q**
- Quicksort, 129, 507–508
  - Quotient, 183
- R**
- Radio communication, CD8.3:8–9
  - Radix Sort, 507–508
  - RAID (redundant arrays of inexpensive disks)
    - big-interleaved parity (RAID 3), 576–577
    - block-interleaved parity (RAID 4), 577–578
    - distributed block-interleaved parity (RAID 5), 578
    - error detecting and correcting code (RAID 2), 575
    - historical development of, CD8.11:5–6
    - mirroring (RAID 1), 575
    - no redundancy (RAID 0), 575
    - P + Q redundancy (RAID 6), 578
    - summary of, 578–579
    - use of term, 574–574

- Random access memory (RAM), 20
  - Raster cathode ray tubes (CRTs), 18
  - Raster refresh buffer, 18
  - Rau, Bob, CD6.13:4
  - Read only memory (ROM), B14, 16, C13–19
  - Read/write head, 23
  - Reals, 189
  - Receive message routine, CD9.1:6
  - Recursive procedures, A26, 29
  - Reduced instruction set computer (RISC), CD2.19:4
    - addressing modes and instruction formats, D5–9
    - Alpha, D27–28
    - architecture, CD5.12:3
    - ARM, D36–38
    - desktop versus embedded, D3–5
    - digital signal-processing extensions, D19
    - MIPS, D9–16, 20–24
    - MIPS16, D41–43
    - MIPS64, D25–27
    - M32R, D40–41
    - multimedia extensions, D16–19
    - PA-RISC 2.0, D34–36
    - PowrPC, D32–33
    - SPARCv.9, D29–32
    - SuperH, D39–40
    - Thumb, D38–39
  - Redundancy. *See* RAID (redundant arrays of inexpensive disks)
  - Reference bit, 519
  - Refresh rate, 18
  - reg, B21–22
  - Register addressing, 100
  - Register file, 293–294, B49, 53–55
    - read, 385, 390, 392, 402
  - Registers, 52–53, 59, 88, 290, 532
    - allocation, CD2.12:7–9
    - architectural, 448
    - dedicated, CD2.19:2
    - destination, 64
    - general-purpose, 135, 138, CD2.19:2–3
    - global pointer, 85
    - IA-32, 137–138
    - jump, 76, 80
    - mapping into numbers, 60–68
    - number, 294
    - renaming, 439
    - special-purpose, CD2.19:2
    - spilling, 58, 80
  - Register use conventions, A22–33
  - Relational databases, CD8.11:4–5
  - Reliability, 573
  - Relocation, 513
  - Relocation information, A13
  - Remainder, 183
  - Remington-Rand, CD1.7:4
  - Remote access times, CD9.1:7
  - Reorder buffer, 443
  - Reproducibility, 255–256
  - Requested word first, 482
  - Reservation stations, 443
  - Response time, 242, 244
  - Restartable instruction, 533
  - Restorations, 572
  - Return address, 80
  - Rings, CD7.9:7
  - Ring topology, CD9.6:27
  - Ripple carry, B39, 44–45
  - RISC. *See* Reduced instruction set computer
  - Ritchie, Dennis, CD2.19:7, CD7.9:8, 11
  - Rotational delay, 570
  - Rotational latency, 570
  - Rounding, 214–215, CD3.10:2–4
  - Routers, CD8.3:6
  - R-type instructions, 292–293, 298
- S**
- Sandisk Corp., 605
  - Scanning, CD2.12:1
  - Scientific notation, 189, 191
  - Secondary memory, 23
  - Sectors, 569
  - Seek, 569
  - Seek time, 569–570
  - Segmentation, 514–515
  - Selector value, B9
  - Selinger, Patricia, CD8.11:5
  - Semantic analysis, CD2.12:1
  - Semaphores, CD9.3:18
  - Semiconductor, 29
  - Send message routine, CD9.1:6
  - Sensitivity list, B24
  - Separate compilation, A18
  - Sequential elements, 290
  - Sequential logic, B5, 55–57
  - Servers, 5
  - Set associative cache, 497, 504
  - set on less than, 75, 77, 165, 301
  - set on less than immediate, 77, 165
  - set on less than immediate unsigned, 165, 169
  - set on less than unsigned, 165, 169
  - Set-up time, B53
  - Shadowing, 575
  - Shared memory, CD9.4:22, 24
  - Shared-memory processors, CD9.1:4–5
  - Shared virtual memory, CD9.4:24
  - Shift amount, 69
  - Shifts, 69
  - Sign and magnitude, 162, 191
  - Sign bit, 163
  - Signed division, 187–188
  - Signed multiplication, 180
  - Signed numbers, 160–170
  - Sign extension, 164, 167–168, 294, 296
  - Significand, 193
  - Silicon, 29
  - Silicon crystal ingot, 29
  - Silicon Graphics. *See* MIPS
  - Simple programmable logic devices (SPLDs), B77
  - Simplicity, 285
  - Simputer, 45
  - Simula-67, CD2.19:7
  - Simultaneous multithreading (SMT), CD9.7:31–34
  - Single address space multiprocessors, CD9.1:4–6
  - Single bus, multiprocessors connected by a, CD9.3:11–20

- Single-cycle implementation scheme, 300–318
  - pipelined performance versus, 372–374
- Single instruction multiple data (SIMD), CD9.11:47–49, 51
- Single instruction single data (SISD), IMD 2.12, CD9.11:47
- Single precision, 192
- Small computer systems interface (SCSI), 573
- Smalltalk, CD2.19:7
- Smith, Jim, CD6.13:2
- Snooping cache coherency, CD9.3:13
- Software
  - applications, 11–12
  - performance affected by, 10
  - systems, 11
  - third-party of shrink-wrap, 5
- sort
  - body for for loop, 126–127
  - code for the body of, 124–126
  - full procedure, 127–128
  - Java, CD2.14:6–14
  - passing parameters, 127
  - preserving registers, 127
  - register allocation, 123
- Source language, A6
- SPARCV.9, D29–32
- Spatial locality, 468–469
- SPEC (System Performance Evaluation Corp.)
  - CPU benchmarks, 254–255, 259–266, CD4.7:2–3, IMD4:7–8
  - file server benchmarks, 599
  - Web server benchmarks, 599
- SPEC ratio, 259
- Speculation, 434–435
- SPECweb99 benchmark, 262–266
- Speedup, IMD4:5
- Spilling registers, 58, 80
- Spilt caches, 487
- SPIM, A40–45, CDA:1–2
  - command-line options, A42, CDA:1–3
- Spin waiting, CD9.3:19, 20
- Split transaction protocol, 585
- SRAM. *See* Static random access memory
- SRT division, 188
- Stack, 80
  - allocating space for data on, 86
  - instructions, CD2.19:3–4, IMD2:8–9
  - pointer, 80
  - segment, A22
- Stale data problem, 595
- Stallman, Richard, CD2.19:8
- Standby spares, 579
- Stanford DASH multiprocessor, CD9.11:52
- State elements, 289–290, B47–48
- Static data segment, 87, A20–22
- Static multiple issue, 433, 435–442, CD6.13:4
- Static random access memory (SRAM), 20, 469, B57–60
- Static storage class, 85
- Stewart, Robert G., CD3.10:7
- Sticky bit, 215
- Stone, Harold S., CD3.10:7
- Stonebraker, Mike, CD8.11:5
- Stop, 440
- Storage
  - for digital cameras, 603–606
  - disk, 569–580, CD8.11:1–4
- Storage classes, types of, 85
- store, 57
- Store buffer, 445, 485
- store byte, 91
- store conditional, CD9.3:19–20
- Stored-program concept, 49, 215
- store half, 94
- store word, 57–59, 294, 300–318
- Strength reduction, 118
- Stretch computer, CD6.13:1–2
- Strings
  - C, 92–93
  - Java, 93–95
- Striping, 575
- Stroustrup, Bjarne, CD2.19:7
- Structural hazards, 375
- Structural specification, B21
- Structured Query Language (SQL), CD8.11:4–5
- Subroutines, CD5.7:2
- subtract, 49–51, 301
- Subtraction, 170–176
- subtract unsigned, 172
- Sum of products, B10–12
- Sun Microsystems, CD4.7:2, CD7.9:9
  - SPARCV.9, D29–32
- Supercomputers
  - defined, 5
  - first CD1.7:5
- SuperH, D39–40
- Superscalar processors, 348, 442–445, CD6.13:4
- Supervisor process, 529
- swap
  - code for the body of, 122–123
  - full procedure, 123
  - Java, CD2.14:6–14
  - register allocation, 122
  - space, 517
- Switched networks, CD8.3:5
- Switches, CD8.3:7
- Switch statement, 76
- Sybase, CD8.11:5
- Symbol table, 108, A12, 13
- Symmetric multiprocessors (SMPs), CD9.1:6
- Synchronization
  - barrier, CD9.3:15
  - coherency and, CD9.3:18–20
  - defined, CD9.1:5
  - failure, B76
- Synchronizers, B75–77
- Synchronous bus, 582–583
- Synchronous system, B48
- Synthetic benchmarks, CD4.7:1–2, IMD4:11–12
- System call, 529, A43–45
- System CPU time, 245
- System performance, 245
- System R, CD8.11:4, 5
- Systems software, 11

**T**

Tags, cache, 475, 504  
 Tail recursion, IMD2:10–11  
 Target language, A6  
 Taylor, George S., CD3.10:8–9  
 Taylor, Robert, CD7.9:9–10  
 TCP/IP, CD8.3:4, CD8.11:7  
 Temporal locality, 468  
 Terabytes, 5  
 Text segment, 87, A13, 20  
 Thacker, Chuck, CD7.9:8  
 Thinking Machines, CD9.11:52  
 Thompson, Ken, CD7.9:8, 11  
 Thornton, J. E., CD6.13:2  
 Thrashing, 537  
 Thread-level parallelism (TLP),  
 CD9.7:33  
 Three Cs model, 543–545  
 Throughput, 242  
 Thumb, D38–39  
 Time, definitions of, 244  
 Time-sharing systems, CD7.9:7–11  
 Timing methodologies, B72–77  
 Tomasulo, Robert, CD6.13:2, 3  
 Tomasulo's algorithm, CD6.13:2  
 Torvald, Linus, CD7.9:10  
 Tournament branch predictors, 423  
 Trace cache, 349  
 Tracks, 569  
 Traiger, Irving, CD8.11:5  
 Trains, computer controlled, 280–281  
 Transaction processing (TP), 598  
 Transaction Processing Council (TPC), 598  
 Transfer time, 570  
 Transistors, 27, 29  
 Translating microprogram to hardware,  
 C27–31  
 Translation hierarchy for C, 106  
 assembler, 107–108  
 compiler, 107  
 linker, 108–111  
 loader, 112

Translation hierarchy for Java, 114  
 compiler, 114–115  
 Java Virtual Machine, 115  
 Just in Time compiler, 115  
 Translation-lookaside buffer (TLB),  
 522–534, CD7.9:5  
 Transportation, technology and, 280–281  
 Truth tables, 302–303, B5, C5, 14, 15, 16  
 Tucker, Stewart, CD5.12:1–2  
 Turing, Alan, CD1.7:3  
 TVM (transmission voie-machine),  
 280–281  
 Two-level logic, B10–14  
 Two's complement representation, 163  
 Types  
 checking, CD2.12:1  
 examples of, 85

**U**

Ullman, Jeff, CD2.19:8  
 Unconditional branches, 73  
 Undefined instruction, exception detec-  
 tion of, 343  
 Underflow, 192, CD3.10:5  
 Unicode, 93–94  
 Uniform memory access (UMA) multi-  
 processors, CD9.1:6, CD9.4:22  
 Units in the last place (ulp), 215  
 UNIVAC I (Universal Automatic Com-  
 puter), CD1.7:4  
 UNIX  
 development of, CD2.19:7,  
 CD7.9:8–11  
 loader, 112  
 object file for, 108  
 Unmapped, 536  
 Unresolved references, A4  
 Unsigned numbers, 160–170  
 Untaken branch hazards, 381  
 USB, 582, 583  
 Use bit, 519  
 User CPU time, 245

**V**

Valid bit, cache, 476  
 VAX, CD5.12:2–3, CD7.9:9  
 Vectored interrupts, 342  
 Vector processing, CD9.11:49–51  
 Verilog, CD5.8:1–7  
 combinational logic in, B23–25  
 data types and operators, B21–22  
 description of, B20–25  
 MIPS arithmetic logic unit (ALU),  
 B36–38  
 program structure, B23  
 sequential logic, B55–57  
 used to describe and model a pipe-  
 line, CD6.7:1–9  
 Very large scale integrated (VLSIs) cir-  
 cuits, 20, 27–28, 29  
 Very long instruction word (VLIW),  
 CD6.13:4  
 VHDL, B20, 21  
 Virtual address, 512  
 Virtually addressed cache, 527  
 Virtual machine, simulation of, A41–42  
 Virtual memory  
 address translation, 512, 521–524  
 defined, 511  
 design, 514–521  
 implementing protection with,  
 528–530  
 overlays, 511–512  
 page, 512  
 page, placing and find, 515–516  
 page faults, 512, 514, 516–521  
 page offset, 513, 514  
 page table, 515–516  
 reasons for, 511–512  
 translation-lookaside buffer (TLB),  
 522–534  
 write-backs, 521  
 Virtual page number, 513  
 Volatile memory, 23  
 von Neumann, John, 48, CD1.7:1–2, 3,

CD3.10:1–2, 3  
Vyssotsky, Victor, CD2.19:8

## W

Wafers, 29–30  
Wall-clock time, 244  
WARP, CD6.13:5  
Web server benchmarks, 599  
Weighted arithmetic mean, 258  
Whetstone synthetic benchmarks,  
  CD4.7:1–2, IMD4:11–12  
while loop, 74–75, 98–99  
  in Java, CD2.14:3–4, 5–6  
Whirlwind project, CD1.7:4, CD7.9:1  
Wide area networks (WANs), 26,  
  CD8.11:11  
WiFi, 44–45, CD8.3:8  
Wilkes, Maurice, CD1.7:2, CD5.12:1,  
  CD7.9:6  
Wilkinson, James H., CD3.10:2

Windows, 11  
wire, B21–22  
Wired Equivalent Privacy, CD8.3:10  
Wireless local area networks (WLANs),  
  CD8.3:8–10  
Wireless technology, 27, 156–157  
Wirth, Niklaus, CD2.19:6  
Wong, Gene, CD8.11:5  
Word, in MIPS architecture, 52  
Working set, 537  
Workload, 254  
World Wide Web, CD8.11:7  
Wozniak, Stephen, CD1.7:5  
Write-around, 484  
Write-back, 385, 392, 402, 484–485, 521,  
  542  
Write buffer, 483–484  
Write control signal, 290, 294  
Write invalidate, CD9.3:14, 17  
Writes  
  handling cache, 483–485

  handling virtual memory, 521  
Write-through, 483, 542

## X

Xerox Palo Alto Research Center  
  (PARC), 16, CD1.7:7–8,  
  CD7.9:9–10, CD8.11:7, 8  
xor, IMD2:21–22  
xspim, A42, CDA:1–4

## Y

Yield, 30

## Z

Zip drive, 19, 20, 25  
Zone bit recording (ZBR), 569  
Zuse, Konrad, CD1.7:3