

Index

CD information is listed by chapter and section number followed by page ranges (CD3.10:6–9). Page references preceded by a single letter refer to appendices.

- 1-bit ALU, C-26–29
 - adder, C-27
 - CarryOut, C-28
 - illustrated, C-29
 - logical unit for AND/OR, C-27
 - for most significant bit, C-33
 - performing AND, OR, and -addition, C-31, C-33
 - See also* Arithmetic logic unit (ALU)
- 32-bit ALU, C-29–38
 - from 31 copies of 1-bit ALU, C-34
 - with 32 1-bit ALUs, C-30
 - defining in Verilog, C-35–38
 - illustrated, C-36
 - ripple carry adder, C-29
 - tailoring to MIPS, C-31–35
 - See also* Arithmetic logic unit (ALU)
- 32-bit immediate operands, 127–28
- 7090/7094 hardware, CD3.10:6

A

- Absolute references, 137
- Abstractions
 - defined, 20
 - hardware/software interface, 20–21
 - principle, 21
- Accumulator architectures, CD2.20:1
- Accumulators, CD2.20:1
- Acronyms, 8

- Addition, 216–21
 - binary, 216–17
 - floating-point, 238–42, 247
 - operands, 217
 - significands, 238
 - speed, 220
 - See also* Arithmetic
- Address-control lines, D-26
- Addresses
 - 32-bit immediates, 127–33
 - base, 83–4
 - byte, 83
 - defined, 82
 - memory, 90
 - virtual, 479–81, 500
- Addressing
 - 32-bit immediates, 127–33
 - Immediate, 130, 131
 - PC-relative, 111, 131
 - register, 130
 - x86 modes, 164–165
- Address select logic, D-24, D-25
- Address space, 478, 482
 - extending, 531
 - flat, 531
 - ID (ASID), 496
 - inadequate, CD5.13:5
 - shared, 625–26
 - single physical, 624
 - unmapped, 500
 - virtual, 496
- Address translation
 - AMD Opteron X4, 526
 - defined, 479
 - fast, 488–90
 - Intel Nehalem, 526
 - TLB for, 488–90

- Advanced Technology Attachment (ATA)
 - disks, 563, 599, 600
- AGP, A-9
- Algol-60, CD2.20:6–7
- Aliasing, 494
- Alignment restriction, 84
- All-pairs N-body algorithm, A-65
- Alpha architecture
 - defined, 513
- ALU control, 302–4
 - bits, 303
 - logic, D-6
 - mapping to gates, D-4–7
 - truth tables, D-5
 - See also* Arithmetic logic unit (ALU)
- ALU control block, 306
 - defined, D-4
 - generating ALU control bits, D-6
- ALUOp, 302, D-6
 - bits, 303, 304
 - control signal, 306
- AMD64, 162, CD2.20:5
- Amdahl's law, 463, 621
 - corollary, 52
 - defined, 51
 - fallacy, 670
- AMD Opteron X4 (Barcelona), 20, 44–50, 286
 - address translation, 526
 - architectural registers, 390
 - base versus fully optimized-performance, 669
 - caches, 527
 - characteristics, 663
 - CPI, miss rates, and DRAM accesses, 528
 - defined, 663
 - illustrated, 662
 - LBMHD performance, 668

- AMD Opteron X4 (Barcelona) (*continued*)
 - memory hierarchies, 526–29
 - microarchitecture, 390, 391
 - miss penalty reduction techniques, 527–29
 - pipeline, 390–92
 - pipeline illustration, 392
 - roofline model, 664
 - shared L3 cache, 529
 - SPEC CPU benchmark, 48–49
 - SPEC power benchmark, 49–50
 - SpMV performance, 681
 - TLB hardware, 526
- American Standard Code for Information Interchange. *See* ASCII
- AND gates, C-12, D-7
- AND operation, 101, C-6
- Annual failure rate (AFR), 559, 599
- Antidependence, 383
- Antifuse, C-78
- Apple computer, CD1.10:6–7
- Application binary interface (ABI), 21
- Application programming interfaces (APIs)
 - defined, A-4
 - graphics, A-14
- Architectural registers, 390
- Arithmetic, 214–68
 - addition, 216–21
 - division, 226–31
 - floating point, 232–57
 - for multimedia, 218–19
 - multiplication, 220–26
 - subtraction, 216–21
- Arithmetic instructions
 - logical, 294
 - operands, 80
 - See also* Instructions
- Arithmetic intensity, 654
- Arithmetic logic unit (ALU)
 - 1-bit, C-26–29
 - 32-bit, C-29–38
 - before forwarding, 354
 - branch datapath, 298
 - hardware, 218
 - memory-reference instruction use, 287
 - for register values, 294
 - R-format operations, 296
 - signed-immediate input, 357
 - See also* ALU control; Control units
- ARM assembler, B1-49
- AREA directive, B1-50
- Armasm directives, B1-53
- ALIGN, B1-53
 - AREA, B1-53–54
 - ASSERT, B1-54
 - CN, B1-54
 - CODE16, CODE32, B1-54
 - CP, B1-54
 - DATA, B1-54–55
 - DCB/DCD/DCI/DCQ/DCW, B1-55
 - ELSE, B1-57
 - END, B1-55
 - ENDFUNC, B1-55, B1-56
 - ENDIF, B1-55, B1-57
 - ENTRY, B1-55
 - EQU, B1-56
 - EXPORT, B1-56
 - EXTERN, IMPORT, B1-56
 - FIELD, B1-56, B1-59
 - FUNCTION, B1-56
 - GBLA, GBLL, GBLS, B1-57
 - GET, B1-57
 - GLOBAL, B1-57
 - IF, B1-57
 - IMPORT, B1-57
 - INCLUDE, B1-57
 - INFO, B1-57
 - KEEP, B1-58
 - LCLA, LCLL, LCLS, B1-58
 - ITORG, B1-58
 - MACRO, MEXIT, MEND, B1-58
 - MAP, B1-59
 - Memory initialization directives, B1-55
 - NCBIN, B1-57
 - NOFF, B1-59
 - OPT, B1-59
 - PROC, B1-60
 - RLIST, RN, B1-60
 - ROUT, B1-60
 - SETA, SETL, SETS, B1-60
 - SPACE, B1-60
 - WHILE, WEND, B1-60
- assembly line, formats, B1-49
- END directive, B1-50
- expressions, B1-51
 - assembly time, B1-51
 - predefined, B1-52–53
- unary and binary operators, B1-52
- label definition, B1-51
- predefined variables, B1-53
- variables, B1-50–51
- types, B1-50
- ARM11 cycle timings, B3-14
- ARM (MIPS) instructions, 156–60, B1-3, B2-3
 - addressing modes, 158
 - ADR instruction, B1-10
 - Arithmetic shift right, B1-11
 - 32-bit program status registers, B2-11
 - Branch with exchange, B1-13
 - Branch with link and exchange, B1-13
 - Breakpoint instruction, B1-12
 - brief history, CD2.20:4
 - Change processor state, B1-16
 - Compare two 32-bit integers, B1-15
 - Coprocessor-dependent operation, B1-14
 - Count leading zeros, B1-14
 - Cpsr/spsr, B1-25
 - calculations, 158–60
 - compare and conditional branch, 158–59
 - condition field, 369
 - data transfer, 157
 - decode table, B2-4–7
 - Cpsr and spsr, B2-11
 - Mode, B2-7
 - Shift, shift size, and Rs, B2-8
 - decoding procedure, B2-3
 - encodings, B2-3
 - formats, 159
 - GNU assemblers, B1-3
 - Instruction types, B1-4
 - logical, 159
 - MIPS (ARM) similarities, 157
 - Rd and Rn, B1-4
 - register-register, 157
 - Relative branch with link, B1-12
 - Rn/Rm, B1-9
 - Syntax, B1-4
 - Condition codes and flags, B1-5–7
 - Optional expressions, B1-4
 - Register names, B1-5
 - Shift operations, B1-7–8
 - Values stored, B1-5
 - Thumb instructions, B1-8
 - Alphabetical list, B1-8–49
- ARPANET, CD6.14:7

- Arrays
 - logic elements, C-18–19
 - multiple dimension, 254
 - pointers versus, 152–56
 - procedures for setting to zero, 153
 - ASCII
 - binary numbers versus, 123
 - character representation, 122
 - defined, 122
 - symbols, 126
 - Assemblers, 136–37
 - defined, 11
 - function, 136
 - microcode, D-30
 - number acceptance, 136
 - object file, 136–37
 - Assembly language
 - defined, 11, 136
 - floating-point, 248
 - illustrated, 12
 - MIPS (ARM), 78, 97–98
 - programs, 135
 - translating into machine language, 97–98
 - Asserted signals, 291, C-4
 - Associativity
 - in caches, 468–69
 - degree, increasing, 467, 504
 - floating-point addition, testing, 258–59
 - increasing, 472–73
 - set, tag size versus, 472–73
 - Asynchronous interconnect, 569
 - Atomic exchange, 134
 - Atomic memory operation, A-21
 - Attribute interpolation, A-43–44
 - Availability, 559
 - Average memory access time (AMAT), 464
 - calculating, 464–65
 - defined, 464
- B**
- Backplane bus, 568
 - Backups, 601–602
 - Bandwidth
 - bisection, 647
 - external to DRAM, 460
 - I/O, 604
 - L2 cache, 661
 - memory, 457, 458
 - network, 647
 - Barrier synchronization, A-18
 - defined, A-20
 - for thread communication, A-34
 - Base addressing, 82–3
 - Base registers, 83
 - Basic block, 107–8
 - Benchmarks
 - defined, 48
 - I/O, 582–84
 - Linpack, 650, CD3.10:3
 - multicores, 643–70
 - multiprocessor, 650–52
 - NAS parallel, 652
 - parallel, 651
 - PARSEC suite, 652
 - SPEC CPU, 48–49
 - SPEC power, 49–50
 - SPECrate, 650
 - SPLASH/SPLASH 2, 650–52
 - Stream, 661
 - Biased notation, 93, 236
 - Big-endian byte order, 84
 - Binary digits. *See* Bits
 - Binary numbers
 - ASCII versus, 123
 - conversion to decimal numbers, 89
 - conversion to hexadecimal numbers, 95
 - defined, 86
 - Bisection bandwidth, 647
 - Bit error rate (BER), CD6.11:9
 - Bit-interleaved parity, 588
 - Bit maps, 17
 - defined, 16, 87
 - goal, 17
 - storing, 17
 - Bits
 - ALUOp, 303, 304
 - defined, 11
 - dirty, 487
 - done, 574
 - error, 574
 - guard, 254–55
 - patterns, 256
 - reference, 485
 - rounding, 255
 - sign, 89
 - state, D-8
 - sticky, 255
 - valid, 444
 - Blocking assignment, C-24
 - Block-interleaved parity, 588–89
 - Blocks
 - combinational, C-4
 - defined, 440
 - finding, 505–506
 - flexible placement, 465–70
 - least recently used (LRU), 471
 - locating in cache, 470–71
 - miss rate and, 451
 - multiword, mapping addresses to, 449–50
 - placement locations, 504–505
 - placement strategies, 467
 - replacement selection, 471
 - replacement strategies, 506–507
 - spatial locality exploitation, 450
 - state, C-4
 - valid data, 444
 - Boolean algebra, C-6
 - Bounds check shortcut, 109
 - Branch datapath
 - ALU, 298
 - operations, 297
 - Branch delay slots
 - defined, 367
 - scheduling, 368
 - Branch equal, 363
 - Branches
 - compiler creation, 106
 - condition, 299
 - decision, moving up, 363
 - delayed, , 299, 329, 363–65, 367, 368
 - ending, 107
 - execution in ID stage, 364
 - pipelined, 364
 - target address, 364
 - unconditional, 106
 - See also* Conditional branches
 - Branch hazards. *See* Control hazards
 - Branch history tables. *See* Branch prediction, buffers
 - Branch-on-equal instruction, 312
 - Branch prediction
 - buffers, 366, 367
 - as control hazard solution, 328
 - defined, 327
 - dynamic, 327, 328, 366–69
 - static, 379

Branch predictors (*continued*)

- accuracy, 367
- correlation, 369
- information from, 368
- tournament, 369

Branch taken

- cost reduction, 363
- defined, 297

Branch target

- addresses, 296
- buffers, 369

Bubbles, 360

Bubble Sort, 151

Bus-based coherent multiprocessors,
CD7.14:6

Buses, 570, 571

- backplane, 568
- defined, C-19
- processor-memory, 568
- synchronous, 569

Bytes

- addressing, 84
- order, 84

C

Cache-aware instructions, 533

Cache coherence, 520–24

- coherence, 520
- consistency, 521
- enforcement schemes, 522
- implementation techniques,
CD5.9:10–11
- migration, 522
- problem, 520, 521, 524
- protocol example, CD5.9:11–15
- protocols, 522
- replication, 522
- snooping protocol, 522–524
- snoopy, CD5.9:16
- state diagram, CD5.9:15

Cache coherency protocol,

- CD5.9:11–15
- finite-state transition diagram,
CD5.9:12, CD5.9:14
- functioning, CD5.9:12
- mechanism, CD5.9:13
- state diagram, CD5.9:15
- states, CD5.9:11–12
- write-back cache, CD5.9:12

Cache controllers, 524

cache coherency protocol,

- CD5.9:11–15

coherent cache implementation

- techniques, CD5.9:10–11

implementing, CD5.9:1–16

snoopy cache coherence, CD5.9:16

SystemVerilog, CD5.9:1–9

Cache hits, 494

Cache misses

- block replacement on, 506–507
- capacity, 509
- compulsory, 509
- conflict, 509
- defined, 451
- direct-mapped cache, 468
- fully associative cache, 469
- handling, 451–52
- memory-stall clock cycles, 461
- reducing with flexible block
placement, 465–70
- set-associative cache, 468–69
- steps, 452
- in write-through cache, 453

Cache performance, 461–78

- calculating, 463
- hit time and, 464
- impact on processor
performance, 462–63

Caches, 443–61

- accessing, 445–51
- associativity in, 468–69
- bits in, 449
- bits needed for, 446
- contents illustration, 447
- defined, 20, 443
- direct-mapped, 443, 445, 449, 465
- disk controller, 564
- empty, 446
- flushing, 581
- FSM for controlling, 515–25
- fully associative, 465
- GPU, A-38
- inconsistent, 452
- index, 446
- Intrinsity FastMATH example, 454–56
- locating blocks in, 470–71
- locations, 444
- memory system design, 457–60
- multilevel, 461, 473–77
- nonblocking, 527
- physically addressed, 494

physically indexed, 493

physically tagged, 493

primary, 474, 475, 478

secondary, 474, 475, 478

set-associative, 465

simulating, 529–30

size, 448

split, 456

summary, 460–61

tag field, 446

tags, CD5.9:10, CD5.9:11

virtually addressed, 494

virtually indexed, 494

virtually tagged, 494

virtual memory and TLB integration,
490–494

write-back, 453, 454, 507, 508

writes, 452–54

write-through, 453, 454, 507, 508

See also Blocks

Callee, 114, 117

Caller, 114

Capabilities, CD5.13:7

Capacity misses, 509

Carry lookahead, C-38–47

4-bit ALUs using, C-45

adder, C-39

fast, with first level of abstraction,
C-39–40fast, with “infinite” hardware,
C-38–39fast, with second level of -abstraction,
C-40–46

plumbing analogy, C-42, C-43

ripple carry speed versus, C-46

summary, C-46–47

Carry save adders, 225

Cause register, 576

defined, 372

illustrated, 577

CDC 6600, CD1.10:6, CD4.15:2

Central processor unit (CPU)

classic performance equation, 35–37

defined, 19

execution time, 30, 31, 32

performance, 30–32

system, time, 30

time, 461

time measurements, 31

user, time, 30

See also Processors

- Cg pixel shader program, A-15–17
- Channel controllers, 579
- Characters
 - ASCII representation, 122
 - in Java, 126–27
- Chips. *See* Integrated circuits (ICs)
- C++ language, CD2.15:26, CD2.20:7
- C language
 - assignment, compiling into MIPS, 79–80
 - compiling, 156, CD2.15:1–2
 - compiling assignment with registers, 81–82
 - compiling while loops in, 106–7
 - sort algorithms, 152
 - translation hierarchy, 135
 - translation to MIPS assembly language, 79
 - variables, 119
- Classes
 - defined, CD2.15:14
 - packages, CD2.15:20
- Clock cycles
 - defined, 31
 - memory-stall, 461, 462
 - number of registers and, 80
 - worst-case delay and, 316
- Clock cycles per instruction (CPI), 33–34, 327
 - one level of caching, 474
 - two levels of caching, 474
- Clocking methodology, 291–3, C-48
 - defined, 291
 - edge-triggered, 291, 292, C-48, C-73
 - level-sensitive, C-74, C-75–76
 - for predictability, 291
- Clock rate
 - defined, 31
 - frequency switched as
 - function of, 40
 - power and, 39
- Clocks, C-48–50
 - edge, C-48, C-50
 - in edge-triggered design, C-73
 - skew, C-74
 - specification, C-57
 - synchronous system, C-48–49
- Clusters, CD7.14:7–8
 - defined, 618, 627, CD7.14:7
 - drawbacks, 628
 - isolation, 630
 - organization, 617
 - overhead in division of memory, 628
 - scientific computing on, CD7.14:7
- Cm, CD7.14:3–4
- C.mmp, CD7.14:3
- Coarse-grained multithreading, 631–32
- Cobol, CD2.20:6
- Code generation, CD2.15:12
- Code motion, CD2.15:6
- Combinational blocks, C-4
- Combinational control units, D-4–8
- Combinational elements, 290
- Combinational logic, 292, C-3, C-9–20
 - arrays, C-18–19
 - decoders, C-9
 - defined, C-5
 - don't cares, C-17–18
 - multiplexors, C-10
 - ROMs, C-14–16
 - two-level, C-11–14
 - Verilog, C-23–26
- Commands, to I/O devices, 574–75
- Commercial computer development, CD1.10:3–9
- Commit units
 - buffer, 385
 - defined, 385
 - in update control, 388
- Common case fast, 172
- Common subexpression elimination, CD2.15:5
- Communication, 24–25
 - overhead, reducing, 43
 - thread, A-34
- Compact code, CD2.20:3
- Compact disks (CDs), 23, 24
- Comparisons, 108–9
 - signed versus unsigned, 108–9
- Compilers, 135
 - branch creation, 106
 - brief history, CD2.20:8
 - conservative, CD2.15:5–6
 - defined, 11
 - front end, CD2.15:2
 - function, 13, 135
 - high-level optimizations, CD2.15:3–4
 - ILP exploitation, CD4.15:4–5
 - Just In Time (JIT), 142
 - optimization, 155, CD2.20:8
 - speculation, 378–79
 - structure, CD2.15:1
- Compiling
 - C assignment statements, 79–80
 - C language, 106–7, 156, CD2.15:1–2
 - floating-point programs, 249–53
 - if-then-else, 105
 - in Java, CD2.15:18–19
 - procedures, 115, 117–18
 - recursive procedures, 117–18
 - while loops, 106–7
- Compressed sparse row (CSR) matrix, A-55, A-56
- Compulsory misses, 509
- Computers
 - application classes, 5–7
 - applications, 4
 - arithmetic for, 214–68
 - characteristics, CD1.10:12
 - commercial development, CD1.10:3–9
 - component organization, 14
 - components, 14, 215, 555
 - design measure, 55
 - desktop, 5, 15
 - embedded, 5–7
 - first, CD1.10:1–3
 - in information revolution, 4
 - instruction representation, 93–100
 - laptop, 18
 - performance measurement, CD1.10:9
 - principles, 99
 - rack mount, 592
 - servers, 5
- Compute Unified Device Architecture.
 - See* CUDA programming environment
- Conditional branches
 - ARM, 158
 - changing program counter with, 369
 - compiling if-then-else into, 105–6
 - defined, 105
 - implementation, 113
 - in loops, 111
 - PC-relative addressing, 111
- Conditional move instructions, 369
- Condition field, 369
- Conflict misses, 509
- Constant memory, A-40

- Constant operands, 85–86
 - frequent occurrence, 86
 - Content Addressable Memory (CAM), 471
 - Context switch, 496
 - Control
 - ALU, 302–4
 - challenge, 370
 - finishing, 313
 - forwarding, 352
 - FSM, D-8–21
 - implementation, optimizing, D-27–28
 - for jump instruction, 315
 - mapping to hardware, D-2–32
 - memory, D-26
 - organizing, to reduce logic, D-31–32
 - pipelined, 345–49
 - Control flow graphs, CD2.15:8–9
 - defined, CD2.15:8
 - illustrated examples, CD2.15:8, CD2.15:9
 - Control functions
 - ALU, mapping to gates, D-4–7
 - defining, 307
 - PLA, implementation, D-7, D-20–21
 - ROM, encoding, D-18–19
 - for single-cycle implementation, 313
 - Control hazards, 325–29, 361–70
 - branch delay reduction, 363–65
 - branch not taken assumption, 363
 - branch prediction as solution, 328
 - defined, 325, 362
 - delayed decision approach, 329
 - dynamic branch prediction, 366–69
 - logic implementation in Verilog, CD4.12:7–9
 - pipeline stalls as solution, 326
 - pipeline summary, 369–70
 - simplicity, 362
 - solutions, 326
 - static multiple-issue processors and, 380
 - Control lines
 - asserted, 309
 - in datapath, 306
 - execution/address calculation, 347
 - final three stages, 347
 - instruction decode/register file read, 347
 - instruction fetch, 347
 - memory access, 348
 - setting of, 307, 309
 - values, 346
 - write-back, 348
 - Control signals
 - ALUOp, 306
 - defined, 292
 - effect of, 307
 - multi-bit, 308
 - pipelined datapaths with, 345
 - truth tables, D-14
 - Control units, 289
 - address select logic, D-24, D-25
 - combinational, implementing, D-4–8
 - with explicit counter, D-23
 - illustrated, 308
 - logic equations, D-11
 - main, designing, 304–312
 - as microcode, D-28
 - MIPS, D-10
 - next-state outputs, D-10, D-12–13
 - output, 302–3, D-10
 - See also* Arithmetic logic unit (ALU)
 - Cooperative thread arrays (CTAs), A-30
 - Coprocessors
 - defined, 253
 - Copy back. *See* Write-back
 - Core MIPS instruction set, 268
 - abstract view, 288
 - implementation, 286–289
 - implementation illustration, 290
 - overview, 287–9
 - subset, 286–7
 - See also* MIPS
 - Cores
 - defined, 41
 - number per chip, 42
 - Correcting code, 588
 - Correlation predictor, 369
 - Cosmic Cube, CD7.14:6
 - Cray computers, CD3.10:4, CD3.10:5
 - Critical word first, 451
 - Crossbar networks, 648
 - CTSS (Compatible Time-Sharing System), CD5.13:8
 - CUDA programming environment, 645, A-5, CDA.11:5
 - barrier synchronization, A-18, A-34
 - defined, A-5
 - development, A-17, A-18
 - hierarchy of thread groups, A-18
 - kernels, A-19, A-24
 - key abstractions, A-18
 - paradigm, A-19–23
 - parallel plus-scan template, A-61
 - per-block shared memory, A-58
 - plus-reduction implementation, A-63
 - programs, A-6, A-24
 - scalable parallel programming with, A-17–23
 - shared memories, A-18
 - threads, A-36
- ## D
- Databases
 - brief history, CD6.14:4
 - Integrated Data Store (IDS), CD6.14:4
 - relational, CD6.14:5
 - Datcenters, 5
 - Data flow analysis, CD2.15:8
 - Data hazards, 322–25, 349–61
 - defined, 322
 - forwarding, 322, 349–61
 - load-use, 324, 363
 - stalls and, 357–60
 - See also* Hazards
 - Data-level parallelism, 635
 - Data parallel problem decomposition, A-17, A-18
 - Datapath elements
 - defined, 293
 - sharing, 299
 - Datapaths
 - branch, 297, 298
 - building, 293–302
 - control signal truth tables, D-14
 - control unit, 308
 - defined, 19
 - design, 293
 - exception handling, 373
 - for fetching instructions, 295
 - for hazard resolution via forwarding, 356
 - for jump instruction, 315
 - for memory instructions, 300

- for MIPS architecture, 301
- in operation for branch-on-equal instruction, 312
- in operation for load instruction, 311
- in operation for R-type instruction, 310
- operation of, 307–312
- pipelined, 330–44
- for R-type instructions, 300, 309
- single, creating, 299–302
- single-cycle, 331
- static two-issue, 381
- Data race, 133
- Data rate, 582
- Data selectors, 289
- Data structure compression, 666
- Data transfer instructions
 - defined, 82
 - load, 82
 - offset, 83
 - store, 84
 - See also* Instructions
- Deasserted signals, 291, C-4
- DEC disk drive, CD6.14:3
- Decimal numbers
 - binary number conversion to, 89–90
 - defined, 86
- Decision-making instructions, 104–13
- Decoders, C-9
 - defined, C-9
 - two-level, C-65
- DEC PDP-8, CD1.10:5
- Deep Web, CD6.14:8
- Delayed branches
 - as control hazard solution, 329
 - defined, 299
 - for five-stage pipelines, 368
 - reducing, 363–65
 - scheduling limitations, 367
 - See also* Branches
- Delayed decision, 329
- DeMorgan's theorems, C-11
- Denormalized numbers, 257
- Dependencies
 - bubble insertion and, 360
 - detection, 351
 - name, 383
 - between pipeline registers, 353
 - between pipeline registers and ALU inputs, 352
 - sequence, 349
- Design
 - compromises and, 172
 - datapath, 293
 - digital, 392–93
 - I/O system, 584–85
 - logic, 289–7, C-1–79
 - main control unit, 304–312
 - memory hierarchy,
 - challenges, 511
 - pipelining instruction sets, 321
- Desktop and server RISCs
 - See also* Reduced instruction set computer (RISC) architectures
- Desktop computers
 - defined, 5
 - illustrated, 15
- D flip-flops, C-51, C-53
- Dicing, 46
- Dies, 46
- Digital design pipeline, 392–93
- Digital video disks (DVDs), 23, 24
- DIMMs (dual inline memory modules), CD5.13:4
- Direct3D, A-13
- Direct-mapped caches
 - address portions, 470
 - choice of, 506
 - defined, 443, 465
 - illustrated, 445
 - memory block location, 466
 - misses, 468
 - single comparator, 471
 - total number of bits, 449
 - See also* Caches
- Direct memory access (DMA)
 - defined, 578
 - multiple devices, 579
 - setup, 579
 - transfers, 579, 581
- Dirty bit, 487
- Dirty pages, 487
- Disk controllers
 - caches, 564
 - defined, 562
 - time, 562
- Disk read time, 563
- Disk storage, 561–65
 - characteristics, 565
 - densities, 563
 - history, CD6.14:1–4
 - interfaces, 563–64
 - as nonvolatile, 561
 - rotational latency, 562
 - sectors, 561
 - seek time, 561
 - tracks, 561
 - transfer time, 562
- Divide algorithm, 229
- Dividend, 226
- Division, 226–31
 - algorithm, 228
 - dividend, 226
 - divisor, 226
 - faster, 231
 - floating-point, 247
 - hardware, 227–28
 - hardware, improved version, 230
 - in ARM, 231
 - operands, 226
 - quotient, 226
 - remainder, 226
 - signed, 229–31
 - SRT, 231
 - See also* Arithmetic
- Divisor, 226
- D latches, C-51, C-52
- Done bit, 574
- Don't cares, C-17–18
 - example, C-17–18
 - term, 304
- Double Data Rate RAMs (DDR RAMs), 459, C-65
- Double precision
 - defined, 234
 - FMA, A-45–46
 - GPU, A-45–46, A-74
 - representation, 237
 - See also* Single precision
- Double words, 165
- Dynamically linked libraries (DLLs), 140–42
 - defined, 140
 - lazy procedure linkage version, 141
- Dynamic branch prediction, 366–69
 - branch prediction buffer, 366
 - defined, 366
 - loops and, 366
 - See also* Control hazards
- Dynamic hardware predictors, 327

Dynamic multiple-issue processors, 378,
383–86
pipeline scheduling, 384–86
superscalar, 383
See also Multiple issue

Dynamic pipeline scheduling,
385–86
commit unit, 385
concept, 386
defined, 384
hardware-based speculation, 386
primary units, 385
reorder buffer, 385
reservation station, 385

Dynamic random access memory
(DRAM), 439, 457, C-63–65
bandwidth external to, 460
cost, 23
defined, 18–19, C-63
DIMM, CD5.13:4
Double Data Rate (DDR), 459
early board, CD5.13:4
GPU, A-37–38
growth of capacity, 27
history, CD5.13:3–4
pass transistor, C-63
SIMM, CD5.13:4, CD5.13:5
single-transistor, C-64
size, 460
speed, 23
synchronous (SDRAM), 459, C-60,
C-65
two-level decoder, C-65

E

Early restart, 451

Edge-triggered clocking methodology,
291, 292, C-48, C-73
advantage, C-49
clocks, C-73
defined, C-48
drawbacks, C-74
illustrated, C-50
rising edge/falling edge, C-48

EDSAC (Electronic Delay-Storage
Automatic Calculator), CD1.10:2,
CD5.13:1–2

Eispack, CD3.10:3

Electrically erasable programmable
read-only memory (EEPROM), 567

Elements
combinational, 290
datapath, 293, 299
memory, C-50–58
state, 291, 292, 294, C-48, C-50

Embedded computers
application requirements, 7
design, 6
growth, CD1.10:11–12

Embedded Microprocessor
Benchmark Consortium
(EEMBC), CD1.10:11–12

Embedded RISCs
See also Reduced instruction set
computer (RISC) architectures

Encoding
ARM instruction, 97
defined, D-31
ROM control function, D-18–19
ROM logic function, C-15
x86 instruction, 168–69

ENIAC (Electronic Numerical
Integrator and Calculator),
CD1.10:1, CD1.10:2, CD1.10:3,
CD5.13:1

EPIC, CD4.15:4

Error bit, 574

Error correction, C-65–67

Error detection, 588, C-66

Ethernet, 24, 25, CD6.14:8
defined, CD6.11:5
multiple, CD6.11:6
success, CD6.11:5

Exception enable, 498

Exception program counters
(EPCs), 371
address capture, 376
defined, 372
in restart determination, 371

Exceptions, 370–77
association, 376
datapath with controls for
handling, 373
defined, 371
detecting, 371
event types and, 371
imprecise, 376
interrupts versus, 370–71
in MIPS architecture, 371–72
overflow, 373
PC, 495, 497

pipelined computer example, 374
in pipelined implementation,
372–77
precise, 376
reasons for, 371–72
result due to overflow in add
instruction, 375
saving/restoring stage on, 501

Execute/address calculation
control line, 347
load instruction, 336
store instruction, 338

Execute or address calculation stage,
336, 338

Execution time
CPU, 30, 31, 32
pipelining and, 330
as valid performance measure, 54

Explicit counters, D-23, D-26

Exponents, 233–45

EX stage
load instructions, 336
overflow exception detection, 373
store instructions, 339

F

Failures
disk, rates, 599–600
mean time between (MTBF), 559
mean time to (MTTF), 559, 560,
599, 616
reasons for, 560
synchronizer, C-77

Fallacies
Amdahl's law, 670
assembly language for performance,
170
commercial binary compatibility
importance, 170–71
defined, 51
disk failure rates, 599–600
GPUs, A-72–74, A-75
low utilization uses little power, 52
MTTF, 599
peak performance, 670–71
pipelining, 393
powerful instructions mean higher
performance, 170
right shift, 262–63
See also Pitfalls

- False sharing, 523
- Fast carry
 - with first level of abstraction, C-39–40
 - with “infinite” hardware, C-38–39
 - with second level of abstraction, C-40–46
- Fast Fourier Transforms (FFT), A-53
- Fiber Distributed Data Interface (FDDI), CD6.14:8
- Fibre Channel Arbitrated Loop (FC-AL), CD6.11:11
- Field programmable devices (FPDs), C-78
- Field programmable gate arrays (FPGAs), C-78
- Fields
 - ARM, 95–96
 - defined, 94
 - format, D-31
 - names, 95
- Filebench, 583
- Files, register, 294, 300, C-50, C-54–56
- File server benchmark (SPECFS), 583
- Fine-grained multithreading, 631, 633
- Finite-state machines (FSMs), 515–20, C-67–72
 - control, D-8–22
 - controllers, 518
 - defined, 517, C-67
 - implementation, 517, C-70
 - Mealy, 518
 - Moore, 518
 - for multicycle control, D-9
 - next-state function, 517, C-67
 - output function, C-67, C-69
 - for simple cache controller, 519
 - state assignment, C-70
 - state register implementation, C-71
 - style of, 518
 - synchronous, C-67
 - SystemVerilog, CD5.9:6–9
 - traffic light example, C-68–70
- Fixed-function graphics pipelines, CDA.11:1
- Flash-based removable memory cards, 23
- Flash memory, 566–68
 - brief history, CD6.14:4
 - characteristics, 23, 566
 - defined, 22, 566
 - as EEPROM, 567
 - NAND, CD6.14:4
 - NOR, 567, CD6.14:4
 - wear leveling, 567
- Flat address space, 531
- Flip-flops
 - defined, C-51
 - D flip-flops, C-51, C-53
- Floating point, 232–57
 - ARM instruction frequency for, 268
 - ARM instructions, 247–49
 - assembly language, 248
 - backward step, CD3.10:3–4
 - binary to decimal conversion, 238
 - challenges, 266
 - defined, 232
 - diversity versus portability, CD3.10:2–3
 - division, 247
 - first dispute, CD3.10:1–2
 - form, 233
 - fused multiply add, 256
 - guard digits, 254–55
 - history, CD3.10:1–10
 - IEEE 754 standard, 235, 236
 - immediate calculations, 254
 - operands, 248
 - operands variation in x86, 261
 - overflow, 233
 - packed format, 261
 - precision, 258
 - procedure with two-dimensional matrices, 250–53
 - programs, compiling, 249–53
 - registers, 252
 - representation, 233–38
 - rounding, 254–55
 - sign and magnitude, 233
 - SSE2 architecture, 261–62
 - subtraction, 247
 - underflow, 233
 - units, 255
 - in x86, 259–62
- Floating-point addition, 238–42
 - arithmetic unit block diagram, 243
 - associativity, testing, 258–59
 - binary, 240, 241
 - illustrated, 240
 - instructions, 247
 - steps, 238–39
- Floating-point arithmetic (GPUs), A-41–46
 - basic, A-42
 - double precision, A-45–46, A-74
 - performance, A-44
 - specialized, A-42–44
 - supported formats, A-42
 - texture operations, A-44
- Floating-point multiplication, 242–47
 - binary, 244–46
 - illustrated, 245
 - instructions, 247
 - significands, 244
 - steps, 242–44
- Floating vectors, CD3.10:2
- Flow-sensitive information, CD2.15:14
- Flushing instructions, 363, 364
 - defined, 363
 - exceptions and, 376
- For loops, 152
 - inner, CD2.15:25
 - SIMD and, CD7.14:2
- Format fields, D-31
- Fortran, CD2.20:6
- Forwarding, 349–61
 - ALU before, 354
 - control, 352
 - datapath for hazard resolution, 356
 - defined, 322
 - functioning, 350–51
 - graphical representation, 323
 - illustrations, CD4.12:25–30
 - multiple results and, 325
 - multiplexors, 356
 - pipeline registers before, 354
 - with two instructions, 322–23
 - Verilog implementation, CD4.12:3–5
- Fractions, 233, 234, 235
- Frame buffer, 17
- Front end, CD2.15:2
- Fully associative caches
 - block replacement strategies, 507
 - choice of, 506
 - defined, 465
 - memory block location, 466
 - misses, 469
 - See also* Caches
- Fully connected networks, 647, 648
- Fused-multiply-add (FMA) operation, 256, A-45–46

G

Game consoles, A-9

Gates, C-3, C-8
 AND, C-12, D-7
 defined, C-8
 delays, C-46
 mapping ALU control function to, D-4–7
 NAND, C-8
 NOR, C-8, C-50

Gateways, CD6.11:6

General Purpose GPUs (GPGPUs), 642, A-5, CDA.11:3

General-purpose registers
 architectures, CD2.20:2–3

Generate
 defined, C-40
 example, C-44
 super, C-41

Gigabytes, 23

Global common subexpression
 elimination, CD2.15:5

Global memory, A-21, A-39

Global miss rates, 475

Global optimization, CD2.15:4–6
 code, CD2.15:6
 defined, CD2.15:4
 implementing, CD2.15:7–10

GNU assembler, B1-61
 Assembler directives, B1-61–64
 Assembler macro, B1-63
 ELF format files, B1-64
 Section flags, B1-64

GPU computing
 defined, A-5
 visual applications, A-6–7
 See also Graphics processing units (GPUs)

GPU system architectures, A-7–12
 graphics logical pipeline, A-10
 heterogeneous, A-7–9
 implications for, A-24
 interfaces and drivers, A-9
 unified, A-10–12

Graph coloring, CD2.15:11

Graphics displays
 computer hardware support, 17
 LCD, 16

Graphics logical pipeline, A-10

Graphics processing units (GPUs), 640–46
 as accelerators, 640
 attribute interpolation, A-43–44
 computing, CDA.11:4
 defined, 44, 620, A-3
 driver software, 641
 evolution, A-5, CDA.11:2
 fallacies and pitfalls, A-72–75
 floating-point arithmetic, A-17, A-41–46, A-74
 future trends, CDA.11:5
 GeForce 8-series generation, A-5
 general computation, A-73–74
 General Purpose (GPGPUs), 642, A-5, CDA.11:3
 graphics mode, A-6
 graphics trends, A-4
 history, A-3–4
 logical graphics pipeline, A-13–14
 main memory, 641
 mapping applications to, A-55–72
 memory, 642
 multilevel caches and, 641
 N-body applications, A-65–72
 NVIDIA architecture, 642–45
 parallelism, 641, A-76
 parallel memory system, A-36–41
 performance doubling, A-4
 perspective, 645–46
 programmable real-time, CDA.11:2–3
 programming, A-12–24
 programming interfaces to, 640, A-17
 real-time graphics, A-13
 scalable, CDA.11:4–5
 summary, A-76
 See also GPU computing

Graphics shader programs, A-14–15

Gresham's Law, 268, CD3.10:1

Grids, A-19

Guard digits
 defined, 254
 rounding with, 254–55

H

Half precision, A-42

Halfwords, 126

Handlers
 defined, 499
 TLB miss, 500

Handshaking protocol, 570

Hard disks
 access times, 23
 defined, 22
 diameters, 23
 illustrated, 22
 read-write head, 22

Hardware
 as hierarchical layer, 10
 language of, 11–13
 operations, 77–80
 supporting procedures in, 113–22
 synthesis, C-21
 translating microprograms to, D-28–32
 virtualizable, 513

Hardware-based speculation, 386

Hardware description languages
 defined, C-20
 using, C-20–26
 VHDL, C-20–21
 See also Verilog

Hardware multithreading, 631–34
 coarse-grained, 631–32
 defined, 631
 fine-grained, 631, 633
 options, 632
 simultaneous, 632–34

Harvard architecture, CD1.10:3

Hazard detection units, 358
 functions, 359
 pipeline connections for, 359

Hazards, 321–29
 control, 325–29, 361–70
 data, 322–25, 349–61
 defined, 321
 forwarding and, 357
 structural, 321–22, 338
 See also Pipelining

Heap
 allocating space on, 119–22
 defined, 120

Heterogeneous systems, A-4–5
 architecture, A-7–9
 defined, A-3

Hexadecimal numbers, 94
 binary number conversion to, 95
 defined, 94

High-level languages, 11–13
 benefits, 13
 computer architectures, CD2.20:4
 defined, 12
 importance, 12

- High-level optimizations, CD2.15:3–4
 - Hit rate, 440
 - Hit time
 - cache performance and, 464
 - defined, 441
 - Hit under miss, 527
 - Hold time, C-54
 - Horizontal microcode, D-32
 - Hot-swapping, 591
 - Hubs, CD6.11:6, CD6.11:7
 - Hybrid hard disks, 567
- I**
- IBM 360/85, CD5.13:6
 - IBM 370, CD6.14:2
 - IBM 701, CD1.10:4
 - IBM 7030, CD4.15:1
 - IBM ALOG, CD3.10:6
 - IBM Blue Genie, CD7.14:8–9
 - IBM Cell QS20
 - base versus fully optimized performance, 669
 - characteristics, 663
 - defined, 665
 - illustrated, 662
 - LBMHD performance, 668
 - roofline model, 664
 - SpMV performance, 681
 - IBM Personal Computer, CD1.10:7, CD2.20:5
 - IBM System/360 computers, CD1.10:5, CD3.10:4, CD3.10:5, CD5.13:5
 - IBM z/VM, CD5.13:7
 - ID stage
 - branch execution in, 364
 - load instructions, 335
 - store instruction in, 335
 - IEEE 754 floating-point standard, 234–236, CD3.10:7–9
 - first chips, CD3.10:7–9
 - in GPU arithmetic, A-42–43
 - implementation, CD3.10:9
 - rounding modes, 255
 - today, CD3.10:9
 - See also* Floating point
 - IEEE 802.11, CD6.11:8–10
 - with base stations, CD6.11:9
 - cellular telephony versus, CD6.11:10
 - defined, CD6.11:8
 - Wired Equivalent privacy, CD6.11:10
 - IEEE 802.3, CD6.14:8
 - I-format, 110
 - If statements, 111
 - If-then-else, 105
 - Immediate instructions, 85
 - Imprecise interrupts, 376, CD4.15:3
 - Index-out-of-bounds check, 109
 - Induction variable elimination, CD2.15:6
 - Inheritance, CD2.15:14
 - In-order commit, 386
 - Input devices, 15
 - Inputs, 304
 - Instances, CD2.15:14
 - Instruction count, 35, 36
 - Instruction cycle timings, B3-3
 - abbreviations, B3-4
 - ARM11, B3-15–18
 - ARM11 address calculation cycles, B3-14
 - ARM7TDMI, B3-6
 - ARM7TDMI core, B3-5
 - ARM7TDMI multiplier, B3-5
 - ARM9TDMI core, B3-6–7
 - calculation steps, B3-3-4
 - dynamic predictor, B3-15
 - pipeline behavior statements, B3-5
 - return stack, B3-15
 - static predictor, B3-14
 - tables, B3-3-5
 - Instruction decode/register file read stage
 - control line, 347
 - load instruction, 334
 - store instruction, 338
 - Instruction execution illustrations, CD4.12:16–30
 - clock cycles 1 and 2, CD4.12:20
 - clock cycles 3 and 4, CD4.12:21
 - clock cycles 5 and 6, CD4.12:22
 - clock cycles 7 and 8, CD4.12:23
 - clock cycle 9, CD4.12:24
 - examples, CD4.12:19–24
 - forwarding, CD4.12:25, CD4.12:26–27
 - no hazard, CD4.12:16–19
 - pipelines with stalls and forwarding, CD4.12:25, CD4.12:28–30
 - Instruction fetch stage
 - control line, 347
 - load instruction, 334
 - store instruction, 338
 - Instruction formats
 - ARM, 158
 - defined, 94
 - jump instruction, 314
 - Instruction latency, 394
 - Instruction-level parallelism (ILP)
 - compiler exploitation, CD4.15:4–5
 - defined, 41, 377
 - exploitation, increasing, 388
 - See also* Parallelism
 - Instruction mix, 37, CD1.10:9
 - Instructions, 74–213
 - add immediate, 86
 - addition, 218
 - arithmetic-logical, 294
 - ARM, 156–60assembly, 79
 - basic block, 107–8
 - cache-aware, 533
 - conditional branch, 105
 - conditional move, 369
 - core, 268
 - data transfer, 82
 - decision-making, 104–13
 - defined, 11, 76
 - as electronic signals, 93
 - encoding, 97
 - fetching, 295
 - fields, 95
 - floating-point, 247–49
 - floating-point (x86), 260
 - flushing, 363, 364, 376
 - immediate, 85
 - introduction to, 76–77
 - I/O, 575
 - jump, 110, 114
 - left-to-right flow, 332
 - load, 82
 - load linked, 135
 - logical operations, 100–4
 - memory access, A-33–34
 - memory-reference, 287
 - multiplication, 225
 - nop, 359
 - performance, 33–34
 - pipeline sequence, 358
 - PTX, A-31, A-32
 - representation in computer, 93–100
 - restartable, 499
 - resuming, 502
 - R-type, 294–5
 - store, 84

- Instructions (*continued*)
 - store conditional, 134
 - subtraction, 218
 - thread, A-30–31
 - vector, 638
 - as words, 76
 - x86, 161–69
 - See also* Arithmetic instructions; MIPS; Operands
 - Instruction set architecture
 - ARM, 156–60
 - branch address calculation, 296
 - defined, 21, 54
 - history, 174
 - maintaining, 54
 - protection and, 514–15
 - thread, A-31–34
 - virtual machine support, 513–14
 - Instruction sets
 - ARM, 369
 - design for pipelining, 321
 - NVIDIA GeForce 8800, A-49
 - x86 growth, 171
 - Instructions per clock cycle (IPC), 377
 - Integrated circuits (ICs)
 - cost, 46
 - defined, 26
 - manufacturing process, 45
 - very large-scale (VLSIs), 26
 - See also* specific chips
 - Integrated Data Store (IDS), CD6.14:4
 - Intel IA-64 architecture, CD4.15:4
 - Intel Nehalem
 - address translation for, 526
 - caches, 527
 - die processor photo, 525
 - memory hierarchies, 526–29
 - miss penalty reduction techniques, 527–29
 - TLB hardware for, 526
 - Intel Paragon, CD7.14:7
 - Intel Threading Building Blocks, A-60
 - Intel Xeon e5345
 - base versus fully optimized
 - performance, 669
 - characteristics, 663
 - defined, 663
 - illustrated, 663
 - LBMHD performance, 668
 - roofline model, 664
 - SpMV performance, 681
 - Interference graphs, CD2.15:11
 - Interleaving, 458, 460
 - Interlock cycles, B3-6
 - ARM10E, B3-12
 - ARM10E core, B3-11
 - ARM9E core, B3-9
 - ARM9Erev2, B3-10
 - ARM9TDMI, B3-7
 - ARM9TDMI multiplier, B3-7
 - Intel XScale, B3-12, B3-13
 - Intel XScale multiplier, B3-12
 - StrongARM1, B3-9
 - StrongARM1 core, B3-8
 - StrongARM1 multiplier, B3-8
 - Intermediate addressing, 129, 130
 - Internetworking, CD6.11:1–3
 - Interprocedural analysis, CD2.15:13
 - Interrupt-driven I/O, 575
 - Interrupt enable, 498
 - Interrupt priority levels (IPLs), 576–78
 - defined, 577
 - higher, 578
 - Interrupts
 - defined, 234, 371
 - event types and, 371
 - exceptions versus, 370–71
 - imprecise, 376, CD4.15:3
 - precise, 376
 - vectored, 372
 - Intrinsity FastMATH processor, 454–56
 - caches, 455
 - data miss rates, 456, 470
 - defined, 454
 - read processing, 492
 - TLB, 490
 - write-through processing, 492
 - Inverted page tables, 486
 - I/O, CD6.14:1–8
 - bandwidth, 604
 - chip sets, 572
 - coherence problem for, 581
 - controllers, 579, 601
 - future directions, 604
 - instructions, 575
 - interrupt-driven, 575
 - memory-mapped, 574
 - parallelism and, 585–592
 - performance, 558
 - performance measures, 582–84
 - processor communication, 575–76
 - rate, 582, 596, 597
 - requests, 558, 604
 - standards, 570
 - system performance impact, 585–586
 - systems, 556
 - transactions, 569
 - I/O benchmarks, 582–83
 - file system, 583–84
 - transaction processing, 582–83
 - Web, 583–84
 - See also* Benchmarks
 - I/O devices
 - characteristics, 557
 - commands to, 574–75
 - diversity, 557
 - expandability, 558
 - illustrated, 556
 - interfacing, 572–81
 - maximum number, 603
 - multiple paths to, 604
 - priorities, 576–78
 - reads/writes to, 558
 - transfers, 571, 578–79
 - I/O interconnects
 - function, 569
 - of x86 processors, 570–72
 - I/O systems
 - design, 584–85
 - design example, 595–97
 - history, 604
 - operating system responsibilities
 - and, 573–74
 - organization, 571
 - peak transfer rate, 603
 - performance, 604
 - power evaluation, 597–98
 - weakest link, 584
 - Issue packets, 379
- ## J
- Java
 - bytecode, 142
 - bytecode architecture, CD2.15:16
 - characters in, 126–27
 - compiling in, CD2.15:18–19
 - goals, 142
 - interpreting, 143, 156, CD2.15:14–15

- keywords, CD2.15:20
- method invocation in,
 - CD2.15:19–20
- pointers, CD2.15:25
- primitive types, CD2.15:25
- programs, starting, 142–43
- reference types, CD2.15:25
- sort algorithms, 152
- strings in, 126–27
- translation hierarchy, 142
- while loop compilation in,
 - CD2.15:17–18
- Java Virtual Machine (JVM), 143,
 - CD2.15:15
- Job-level parallelism, 618
- Jump instructions, 298
 - branch instruction versus, 314
 - control and datapath for, 315
 - implementing, 314
 - instruction format, 314
- Just In Time (JIT) compilers,
 - 143, 673

K

- Karnaugh maps, C-18
- Kernel mode, 495
- Kernels
 - CUDA, A-19, A-24
 - defined, A-19

L

- LAPACK, 259
- Laptop computers, 18
- Large-scale multiprocessors,
 - CD7.14:6–7, CD7.14:8–9
- Latches
 - defined, C-51
 - D latch, C-51, C-52
- Latency
 - constraints, 584
 - instruction, 394
 - memory, A-74–75
 - pipeline, 330
 - rotational, 562
 - use, 381, 382
- Lattice Boltzmann Magneto-
 - Hydrodynamics (LBMHD), 666–68
 - defined, 666
 - optimizations, 667–68
 - performance, 668

- Leaf procedures
 - defined, 117
 - example, 126
 - See also* Procedures
- Least recently used (LRU)
 - as block replacement strategy, 507
 - defined, 471
 - pages, 485
- Least significant bits, C-32
 - defined, 87
- Left-to-right instruction flow, 332
- Level-sensitive clocking, C-74, C-75–76
 - defined, C-74
 - two-phase, C-75
- Lines. *See* Blocks
- Linkers, 137–40
 - defined, 137
 - executable files, 137
 - steps, 137
 - using, 138–40
- Linking object files, 138–39
- Linpack, 650, CD3.10:3
- Liquid crystal displays (LCDs), 16
- Live range, CD2.15:10
- Livermore Loops, CD1.10:10
- Load balancing, 623–24
- Loaders, 140
- Load instructions
 - access, A-41
 - base register, 305
 - compiling with, 84
 - datapath in operation for, 311
 - defined, 82
 - EX stage, 336
 - halfword unsigned, 157
 - ID stage, 335
 - IF stage, 335
 - load byte unsigned, 157
 - load half, 126
 - load upper immediate, 160
 - MEM stage, 337
 - pipelined datapath in, 341
 - signed, 123
 - unit for implementing, 297
 - unsigned, 123
 - WB stage, 337
 - See also* Store instructions
- Load-store architectures, CD2.20:2
- Load-use data hazard, 324, 363
- Load-use stalls, 363
- Load word, 82, 84

- Local area networks (LANs), CD6.11:5–8,
 - CD6.14:8
 - defined, 25
 - Ethernet, CD6.11:5–6
 - hubs, CD6.11:6, CD6.11:7
 - routers, CD6.11:6
 - switches, CD6.11:6–7
 - wireless, CD6.11:8–11
 - See also* Networks
- Locality
 - principle, 438, 439
 - spatial, 438–39, 442
 - temporal, 438, 439, 442
- Local memory, A-21, A-40
- Local miss rates, 475
- Local optimization, CD2.15:4–6
 - defined, CD2.15:4
 - implementing, CD2.15:7
 - See also* Optimization
- Locks, 625
- Lock synchronization, 133
- Logic
 - address select, D-24, D-25
 - ALU control, D-6
 - combinational, 292, C-5, C-9–20
 - components, 291
 - control unit equations, D-11
 - design, 289–7, C-1–79
 - equations, C-7
 - minimization, C-18
 - programmable array (PAL), C-78
 - sequential, C-5, C-56–58
 - two-level, C-11–14
- Logical operations, 100–104
 - ARM, 158
 - defined, 100–4
 - NOR, 101
 - NOT, 102
 - OR, 101
 - shifts, 102
- Long-haul networks, CD6.11:5
- Long instruction word (LIW),
 - CD4.15:4
- Lookup tables (LUTs), C-79
- Loops, 106–8
 - conditional branches in, 111
 - defined, 106
 - for, 152, CD2.15:25
 - prediction and, 366
 - test, 153, 154
 - while, compiling, 106–7

Loop unrolling
 defined, 383, CD2.15:3
 for multiple-issue pipelines, 383
 register renaming and, 383

M

Machine code, 94
 Machine instructions, 94
 Machine language
 defined, 11, 94
 illustrated, 12
 SRAM, 20
 Magnetic disks. *See* Hard disks
 Magnetic tapes, 601–2
 defined, 23
 use history, 601–2
 Main memory, 479
 defined, 21
 page tables, 487
 physical addresses, 478, 479
 See also Memory
 Mapping applications, A-55–72
 Mark computers, CD1.10:3
 Mealy machine, 518, C-68, C-71, C-72
 Mean time between failures
 (MTBF), 559
 Mean time to failure (MTTF), 559, 560
 fallacies, 599
 ratings, 586
 Mean time to repair (MTTR), 559, 560
 Memory
 addresses, 90
 affinity, 666, 681
 atomic, A-21
 bandwidth, 457, 458
 cache, 20, 443–78
 CAM, 471
 constant, A-40
 control, D-26
 defined, 17
 DRAM, 18–19, 439, 457, 459, C-63–65
 efficiency, 628
 flash, 22, 23, 566–68, CD6.14:4
 global, A-21, A-39
 GPU, 642
 instructions, datapath for, 300
 local, A-21, A-40
 main, 21
 nonvolatile, 21
 operands, 81–82

 parallel system, A-36–41
 read-only (ROM), C-14–16
 SDRAM, 459
 secondary, 22
 shared, A-21, A-39–40
 spaces, A-39
 SRAM, C-58–62
 stalls, 464
 technologies for building, 25–26
 texture, A-40
 virtual, 478–503
 volatile, 21
 Memory access instructions, A-33–34
 Memory access stage
 control line, 348
 load instruction, 336
 store instruction, 338
 Memory consistency model, 524
 Memory elements, C-50–58
 clocked, C-51
 D flip-flop, C-51, C-53
 D latch, C-52
 DRAMs, C-63–67
 flip-flop, C-51
 hold time, C-54
 latch, C-51
 setup time, C-53, C-54
 SRAMs, C-58–62
 unclocked, C-51
 Memory hierarchies
 block (or line), 440
 cache performance, 461–78
 caches, 443–61
 common framework, 504–511
 defined, 439
 design challenges, 511
 development, CD5.13:5–7
 exploiting, 436–534
 inclusion, 528
 level pairs, 441
 multiple levels, 440
 overall operation of, 493
 parallelism and, 520–24
 pitfalls, 529–33
 program execution time and, 477
 quantitative design parameters, 504
 reliance on, 441
 structure, 440
 structure diagram, 442
 variance, 477
 virtual memory, 478–503

Memory-mapped I/O
 defined, 574
 Memory-stall clock cycles, 461, 462
 Message passing
 defined, 627
 multiprocessors, 627–31
 Metastability, C-76
 Methods
 defined, CD2.15:14
 invoking in Java, CD2.15:19–20
 Microarchitectures
 AMD Opteron X4 (Barcelona), 391
 defined, 390
 Microcode
 assembler, D-30
 control unit as, D-28
 defined, D-27
 dispatch ROMs, D-30–31
 field translation, D-29
 horizontal, D-32
 vertical, D-32
 Microinstructions, D-31
 Microprocessors
 design shift, 619
 multicore, 8, 41, 618
 Microprograms
 as abstract control representation,
 D-30
 translating to hardware, D-28–32
 Migration, 522
 Million instructions per second
 (MIPS), 53
 Minterms
 defined, C-12, D-20
 in PLA implementation, D-20
 MIP-map, A-44
 MIPS, 78, 97–98
 arithmetic core, 267
 arithmetic instructions, 77
 ARM similarities, 157
 control registers, 497
 control unit, D-10
 exceptions in, 371–72
 instruction classes, 173
 Mirroring, 588
 Miss penalty
 defined, 441
 determination, 450
 multilevel caches, reducing,
 473–77
 reduction techniques, 527–29

- Miss rates
 - block size versus, 451
 - data cache, 505
 - defined, 440
 - global, 475
 - improvement, 450
 - Intrinsity FastMATH
 - processor, 456
 - local, 475
 - miss sources, 510
 - split cache, 456
 - Miss under miss, 527
 - Moore machines, 518, C-68, C-71, C-72
 - Moore's law, 640, A-72–73
 - Most significant bit
 - 1-bit ALU for, C-33
 - defined, 87
 - Motherboards, 17
 - Mouse anatomy, 16
 - MS-DOS, CD5.13:10–11
 - Multicore multiprocessors, 41
 - benchmarking with roofline model, 661–70
 - characteristics, 663
 - defined, 8, 618
 - system organization, 662
 - two sockets, 662
 - MULTICS (Multiplexed Information and Computing Service), CD5.13:8–9
 - Multilevel caches
 - complications, 475
 - defined, 461, 475
 - miss penalty, reducing, 473–77
 - performance of, 473–74
 - summary, 477–78
 - See also* Caches
 - Multimedia arithmetic, 218–19
 - Multimedia extensions
 - vector versus, 639
 - Multiple-clock-cycle pipeline
 - diagrams, 342
 - defined, 342
 - five instructions, 343
 - illustrated, 343
 - Multiple dimension arrays, 254
 - Multiple instruction multiple data (MIMD), 645
 - defined, 634
 - first multiprocessor, CD7.14:3
 - Multiple instruction single data (MISD), 635
 - Multiple issue, 377–86
 - code scheduling, 382
 - defined, 377
 - dynamic, 378, 383–86
 - issue packets, 379
 - loop unrolling and, 383
 - processors, 377, 378
 - static, 378, 379–83
 - throughput and, 387
 - Multiplexors, C-10
 - controls, 517
 - in datapath, 306
 - defined, 288
 - forwarding, control values, 356
 - selector control, 300
 - two-input, C-10
 - Multiplicand, 220
 - Multiplication, 220–25
 - fast, hardware, 221
 - faster, 225
 - first algorithm, 222
 - floating-point, 242–46
 - hardware, 221–23
 - instructions, 226
 - multiplicand, 220
 - multiplier, 220
 - operands, 220
 - product, 220
 - sequential version, 221–23
 - signed, 224
 - See also* Arithmetic
 - Multiplier, 220
 - Multiply-add (MAD), A-42
 - Multiply algorithm, 224
 - Multiprocessors
 - benchmarks, 650–52
 - bus-based coherent, CD7.14:6
 - defined, 618
 - historical perspective, 674
 - large-scale, CD7.14:6–7, CD7.14:8–9
 - message-passing, 627–31
 - multithreaded architecture, A-26–27, A-35–36
 - organization, 617, 627
 - for performance, 672–73
 - shared memory, 619, 624–26
 - software, 618
 - TFLOPS, CD7.14:5
 - UMA, 625
 - Multistage networks, 648
 - Multithreaded multiprocessor
 - architecture, A-25–36
 - conclusion, A-36
 - ISA, A-31–34
 - massive multithreading, A-25–26
 - multiprocessor, A-26–27
 - multiprocessor comparison, A-35–36
 - SIMT, A-27–30
 - special function units (SFUs), A-35
 - streaming processor (SP), A-34
 - thread instructions, A-30–31
 - threads/thread blocks management, A-30
 - Multithreading, A-25–26
 - coarse-grained, 631–32
 - defined, 620
 - fine-grained, 631, 633
 - hardware, 631–34
 - simultaneous (SMT), 632–34
 - Must-information, CD2.15:14
 - Mutual exclusion, 133
- ## N
- Name dependence, 383
 - NAND flash memory, CD6.14:4
 - NAND gates, C-8
 - NAS (NASA Advanced Supercomputing), 652
 - N-body
 - all-pairs algorithm, A-65
 - GPU simulation, A-71
 - mathematics, A-65–67
 - multiple threads per body, A-68–69
 - optimization, A-67
 - performance comparison, A-69–70
 - results, A-70–72
 - shared memory use, A-67–68
 - Negation shortcut, 90–91
 - Nested procedures, 117–19
 - compiling recursive procedure showing, 117–18
 - defined, 117
 - Network of Workstations, CD7.14:7–8

Networks, 24–25, 598–99, CD6.11:1–11
 advantages, 24
 bandwidth, 647
 characteristics, CD6.11:1
 crossbar, 648
 fully connected, 647, 648
 local area (LANs), 25, CD6.11:5–8,
 CD6.14:8
 long-haul, CD6.11:5
 multistage, 648
 OSI model layers, CD6.11:2
 peer-to-peer, CD6.11:2
 performance, CD6.11:7–8
 protocol families/suites, CD6.11:1
 switched, CD6.11:5
 wide area (WANs), 25, CD6.14:7–8
 Network topologies, 646–49
 implementing, 648–49
 multistage, 649
 Newton's iteration, 253
 Next state
 nonsequential, D-24
 sequential, D-23
 Next-state function, 517, C-67
 defined, 517
 implementing, with sequencer,
 D-22–28
 Next-state outputs, D-10, D-12–13
 example, D-12–13
 implementation, D-12
 logic equations, D-12–13
 truth tables, D-15
 Nonblocking assignment, C-24
 Nonblocking caches, 389, 527
 Nonuniform memory access
 (NUMA), 625
 Nonvolatile memory, 21
 Nonvolatile storage, 561
 Nops, 359
 NOR flash memory, 567, CD6.14:4
 NOR gates, C-8
 cross-coupled, C-50
 D latch implemented with, C-52
 NOR operation, 101
 North bridge, 570
 NOT operation, 102
 No write allocation, 453
 Numbers
 binary, 87
 computer versus real-world, 257
 decimal, 86, 89

 denormalized, 257
 hexadecimal, 94–95
 signed, 86–90
 unsigned, 86–90
 NVIDIA GeForce 3, CDA.11:1
 NVIDIA GeForce 8800, A-46–55,
 CDA.11:3
 all-pairs N-body algorithm, A-71
 dense linear algebra computations,
 A-51–53
 FFT performance, A-53
 instruction set, A-49
 performance, A-51
 rasterization, A-50
 ROP, A-50–51
 scalability, A-51
 sorting performance, A-54–55
 special function approximation
 statistics, A-43
 special function unit (SFU), A-50
 streaming multiprocessor (SM),
 A-48–49
 streaming processor, A-49–50
 streaming processor array (SPA), A-46
 texture/processor cluster (TPC),
 A-47–48
 NVIDIA GPU architecture, 642–45

O

Object files, 136
 debugging information, 137
 header, 136
 linking, 138–39
 relocation information, 136
 static data segment, 136
 symbol table, 136, 137
 text segment, 136
 Object-oriented languages
 brief history, CD2.20:7
 defined, 156, CD2.15:14
 See also Java
 One's complement, 93, C-29
 Opcodes
 control line setting and, 309
 defined, 95, 305
 OpenGL, A-13
 OpenMP (Open MultiProcessing), 652
 Open Systems Interconnect (OSI) model,
 CD6.11:2
 Operands, 80–86

 32-bit immediate, 127–28
 adding, 217
 arithmetic instructions, 80
 compiling assignment when in
 memory, 82–83
 constant, 85–86
 division, 227
 floating-point, 248, 261
 memory, 81–83
 MIPS, 78
 multiplication, 220
 See also Instructions
 Operating systems
 brief history, CD5.13:8–11
 defined, 10
 disk access scheduling pitfall, 602–3
 encapsulation, 21
 Operations
 hardware, 77–80
 logical, 100–4
 x86 integer, 165–67
 Optical disks
 defined, 23
 technology, 24
 Optimization
 class explanation, CD2.15:13
 compiler, 155
 control implementation, D-27–28
 global, CD2.15:4–6
 high-level, CD2.15:3
 local, CD2.15:4–6, CD2.15:7
 manual, 155
 Out-of-order execution
 defined, 386
 performance complexity, 475
 processors, 389
 Output devices, 15
 Overflow
 defined, 88, 233
 detection, 218
 exceptions, 373
 floating point, 233
 occurrence, 90
 saturation and, 219
 subtraction, 218

P

Packed floating-point format, 261
 Page faults, 484
 for data access, 499

- defined, 479, 480
- handling, 481, 496–502
- virtual address causing, 500
- See also* Virtual memory
- Pages
 - defined, 479
 - dirty, 487
 - finding, 482
 - LRU, 485
 - offset, 480
 - physical number, 480
 - placing, 482
 - size, 481
 - virtual number, 480
 - See also* Virtual memory
- Page tables, 506
 - defined, 482
 - illustrated, 485
 - indexing, 483
 - inverted, 486
 - levels, 486–487
 - main memory, 487
 - register, 483
 - storage reduction techniques, 486–487
 - updating, 482
 - VMM, 515
- Parallelism, 41, 377–89
 - data-level, 635
 - debates, CD7.14:4–6
 - GPUs and, 641, A-76
 - instruction-level, 41, 377, 388
 - I/O and, 585–592
 - job-level, 618
 - memory hierarchies and, 520–24
 - multicore and, 634
 - multiple issue, 377–86
 - multithreading and, 634
 - performance benefits, 43
 - process-level, 618
 - task, A-24
 - thread, A-22
- Parallel memory system, A-36–41
 - caches, A-38
 - constant memory, A-40
 - DRAM considerations, A-37–38
 - global memory, A-39
 - load/store access, A-41
 - local memory, A-40
 - memory spaces, A-39
 - MMU, A-38–39
- ROP, A-41
- shared memory, A-39–40
- surfaces, A-41
- texture memory, A-40
- See also* Graphics processing units (GPUs)
- Parallel processing programs, 620–24
 - creation difficulty, 620–24
 - defined, 618
 - for message passing, 628–29
 - for shared address space, 625–26
 - use of, 672
- Parallel reduction, A-62
- Parallel scan, A-60–63
 - CUDA template, A-61
 - defined, A-60
 - inclusive, A-60
 - tree-based, A-62
- Parallel software, 619
- Paravirtualization, 533
- Parity, 588
 - bit-interleaved, 588
 - block-interleaved, 588–590
 - code, C-65
 - disk, 589
 - distributed block-interleaved, 589–90
- PARSEC (Princeton Application Repository for Shared Memory Computers), 652
- Pass transistor, C-63
- PCI-Express (PCIe), A-8
- PC-relative addressing, 111, 130
- Peak floating-point performance, 654
- Peak transfer rate, 603
- Peer-to-peer networks, CD6.11:2
- Pentium bug morality play, 264–65
- Performance, 26–38
 - assessing, 26–27
 - classic CPU equation, 35–37
 - components, 37
 - CPU, 30–32
 - defining, 27–30
 - equation, using, 34
 - improving, 32–33
 - instruction, 33–34
 - measuring, 30–32, CD1.10:9
 - networks, CD6.11:7–8
 - program, 38
 - ratio, 30
 - relative, 29
- response time, 28, 29
- sorting, A-54–55
- throughput, 28
- time measurement, 30
- Petabytes, 5
- Physical addresses, 479
 - defined, 478
 - mapping to, 480
 - space, 624, 626
- Physically addressed caches, 494
- Physical memory. *See* Main memory
- Pipelined branches, 364
- Pipelined control, 345–49
 - control lines, 346, 347
 - overview illustration, 361
 - specifying, 347
 - See also* Control
- Pipelined datapaths, 330–44
 - with connected control signals, 348
 - with control signals, 345
 - corrected, 341
 - illustrated, 333
 - in load instruction stages, 341
- Pipelined dependencies, 350
- Pipeline registers
 - before forwarding, 354
 - dependences, 352, 353
 - forwarding unit selection, 357
- Pipelines
 - AMD Opteron X4 (Barcelona), 390–92
 - branch instruction impact, 362
 - effectiveness, improving, CD4.15:3–4
 - execute and address calculation stage, 336, 338
 - five-stage, 319, 334–36, 344
 - fixed-function graphics, CDA.11:1
 - graphic representation, 323, 342–44
 - instruction decode and register file read stage, 334, 338
 - instruction fetch stage, 334, 338
 - instructions sequence, 358
 - latency, 330
 - memory access stage, 336, 338
 - multiple-clock-cycle diagrams, 342
 - performance bottlenecks, 388
 - single-clock-cycle diagrams, 342
 - stages, 319
 - static two-issue, 380
 - write-back stage, 336, 338

- Pipeline stalls, 324–25
 - avoiding with code reordering, 324–25
 - data hazards and, 357–60
 - defined, 324
 - insertion, 360
 - load-use, 363
 - as solution to control hazards, 326
- Pipelining, 316–30
 - advanced, 388–89
 - benefits, 317
 - control hazards, 325–14
 - data hazards, 322–25
 - defined, 316
 - exceptions and, 372–77
 - execution time and, 330
 - fallacies, 393
 - hazards, 321–29
 - instruction set design for, 321
 - laundry analogy, 317
 - overview, 316–30
 - paradox, 317
 - performance improvement, 321
 - pitfall, 393–94
 - simultaneous executing instructions, 330
 - speed-up formula, 319
 - structural hazards, 321–22, 338
 - summary, 329
 - throughput and, 330
- Pitfalls
 - address space extension, 531
 - associativity, 531
 - defined, 51
 - GPUs, A-74–75
 - ignoring memory system
 - behavior, 530
 - magnetic tape backups, 601–2
 - memory hierarchies, 529–33
 - moving functions to I/O
 - processor, 601
 - network feature provision, 600–601
 - operating system disk accesses, 602–3
 - out-of-order processor
 - evaluation, 531
 - peak transfer rate performance, 603
 - performance equation subset, 52–53
 - pipelining, 393–94
 - pointer to automatic variables, 171
 - sequential word addresses, 171
 - simulating cache, 529–30
 - software development with
 - multiprocessors, 671
 - VMM implementation, 531–33
 - See also* Fallacies
- Pixel shader example, A-15–17
- Pizza boxes, 593
- Pointers
 - arrays versus, 152–56
 - incrementing, 154
 - Java, CD2.15:25
 - stack, 115, 116
- Polling, 575
- Pop, 115
- Power
 - clock rate and, 39
 - critical nature of, 55
 - efficiency, 388–89
 - relative, 40
- PowerPC
- P + Q redundancy, 590
- Precise interrupts, 376
- Prediction
 - 2-bit scheme, 367
 - accuracy, 366, 367
 - dynamic branch, 366–69
 - loops and, 366
 - steady-state, 366
- Prefetching, 533, 666
- Primary memory. *See* Main memory
- Primitive types, CD2.15:25
- Priority levels, 576–78
- Procedure calls
 - preservation across, 119
- Procedures, 113–22
 - compiling, 115
 - compiling, showing nested
 - procedure linking, 117–18
 - defined, 113
 - execution steps, 113
 - frames, 119
 - leaf, 117
 - nested, 117–18
 - recursive, 121
 - for setting arrays to zero, 153
 - sort, 145–51
 - strcpy, 124–25
 - string copy, 124–25
 - swap, 144–45
- Process identifiers, 496
- Process-level parallelism, 618
- Processor-memory bus, 568
- Processors, 284–395
 - control, 19
 - as cores, 41
 - datapath, 19
 - defined, 14, 19
 - dynamic multiple-issue, 378
 - I/O communication with, 575–76
 - multiple-issue, 377, 378
 - out-of-order execution, 389, 475
 - performance growth, 42
 - ROP, A-12, A-41
 - speculation, 378–79
 - static multiple-issue, 378, 379–83
 - streaming, 643, A-34
 - superscalar, 383, 384, 385–386, 632, CD4.15:4
 - technologies for building, 25–26
 - two-issue, 381
 - vector, 636–39
 - VLIW, 380
- Product, 220
- Product of sums, C-11
- Program counters (PCs), 293
 - changing with conditional
 - branch, 369
 - defined, 110, 293
 - exception, 495, 497
 - incrementing, 293, 295
 - instruction updates, 334
- Programmable array logic (PAL), C-78
- Programmable logic arrays (PLAs)
 - component dots illustration, C-16
 - control function implementation, D-7, D-20–21
 - defined, C-12
 - example, C-13–14
 - illustrated, C-13
 - ROMs and, C-15–16
 - size, D-20
 - truth table implementation, C-13
- Programmable logic devices (PLDs), C-78
- Programmable real-time graphics, CDA.11:2–3
- Programmable ROMs (PROMs), C-14
- Programming languages
 - brief history of, CD2.20:6–7
 - object-oriented, 156
 - variables, 80
 - See also specific languages*

Program performance
 elements affecting, 38
 understanding, 9

Programs
 assembly language, 135
 Java, starting, 142–43
 parallel processing, 620–24
 starting, 135–43
 translating, 135–43

Propagate
 defined, C-40
 example, C-44
 super, C-41

Protected keywords, CD2.15:20

Protection
 defined, 478
 group, 588
 implementing, 494–96
 mechanisms, CD5.13:7
 VMs for, 512

Protocol families/suites
 analogy, CD6.11:2–3
 defined, CD6.11:1
 goal, CD6.11:2

Protocol stacks, CD6.11:3

Pseudoinstructions
 defined, 136

Pthreads (POSIX threads), 652

PTX instructions, A-31, A-32

Public keywords, CD2.15:20

Push
 defined, 115
 using, 117

Q

Quad words, 165

Quicksort, 475, 476

Quotient, 226

R

Race, C-73

Radix sort, 475, 476, A-63–65
 CUDA code, A-64
 implementation, A-63–65

RAID. *See* Redundant arrays of
 inexpensive disks

RAMAC (Random Access Method
 of Accounting and Control),
 CD6.14:1, CD6.14:2

Rank units, 592, 593

Rasterization, A-50

Raster operation (ROP) processors,
 A-12, A-41
 fixed function, A-41
 GeForce 8800, A-50–51

Raster refresh buffer, 17

Read-only memories (ROMs), C-14–16
 control entries, D-16–17
 control function encoding, D-18–19
 defined, C-14
 dispatch, D-25
 implementation, D-15–19
 logic function encoding, C-15
 overhead, D-18
 PLAs and, C-15–16
 programmable (PROM), C-14
 total size, D-16

Read-stall cycles, 462

Receive message routine, 627

Recursive procedures, 121
 clone invocation, 117
See also Procedures

Reduced instruction set computer
 (RISC) architectures
See also Desktop and server RISCs;
 Embedded RISCs

Reduction, 626

Redundant arrays of inexpensive disks
 (RAID), 586–592
 calculation of, 591
 defined, 586
 example illustration, 587
 history, CD6.14:6–7
 PCI controller, 597
 popularity, 586
 RAID 0, 587
 RAID 1, 588, CD6.14:6
 RAID 1 + 0, 592
 RAID 2, 588, CD6.14:6
 RAID 3, 588, CD6.14:6, CD6.14:7
 RAID 4, 588–89, CD6.14:6
 RAID 5, 589–90, CD6.14:6, CD6.14:7
 RAID 6, 590
 spread of, CD6.14:7
 summary, 590–91
 use statistics, CD6.14:7

Reference bit, 485

References
 absolute, 137
 types, CD2.15:25

Register addressing, 130

Register allocation, CD2.15:10–12

Register files, C-50, C-54–56
 in behavioral Verilog, C-57
 defined, 294, C-50, C-54
 single, 300
 two read ports implementation,
 C-55
 with two read ports/one write port,
 C-55
 write port implementation, C-56

Register-memory architecture,
 CD2.20:2

Registers
 architectural, 390
 base, 83
 Cause, 372, 576, 577
 clock cycle time and, 81
 compiling C assignment with, 81
 defined, 80
 destination, 96, 305
 left half, 334
 MIPS conventions, 121
 number specification, 295
 page table, 483
 pipeline, 352, 353, 354, 357
 primitives, 80–81
 renaming, 383
 right half, 334
 spilling, 85
 Status, 372, 576, 577
 temporary, 81, 116
 variables, 81
 x86, 163

Relational databases, CD6.14:5

Relative performance, 29

Relative power, 40

Reliability, 559

Remainder
 defined, 226

Reorder buffers, 385, 388, 389

Replication, 522

Requested word first, 451

Reservation stations
 buffering operands in, 386
 defined, 385

Response time, 28, 29

Restartable instructions, 499

Restorations, 559

Return address, 114

Return from exception (ERET), 495

R-format, 305
 ALU operations, 296
 Ripple carry
 adder, C-29
 carry lookahead speed versus, C-46
 RISC. *See* Desktop and server RISCs;
 Embedded RISCs; Reduced
 instruction set computer (RISC)
 architectures
 Roofline model, 653–61
 benchmarking multicores with,
 661–70
 with ceilings, 658, 660
 computational roofline, 659
 IBM Cell QS20, 664
 illustrated, 655
 Intel Xeon e5345, 664
 I/O intensive kernel, 661
 Opteron generations, 656
 with overlapping areas shaded, 660
 peak floating-point performance, 654
 peak memory performance, 655
 Sun UltraSPARC T2, 664
 with two kernels, 660
 Rotational latency, 562
 Rounding
 accurate, 254
 bits, 255
 defined, 254
 with guard digits, 254
 IEEE 754 modes, 255
 Routers, CD6.11:6
 Row-major order, 253
 R-type instructions, 294–5
 datapath for, 309
 datapath in operation for, 310

S

Saturation, 219
 Scalable GPUs, CDA.11:4–5
 SCALAPAK, 259
 Scaling
 strong, 623, 624
 weak, 623
 Scientific notation
 adding numbers in, 238
 defined, 232
 for reals, 232
 Secondary memory, 22
 Sectors, 561
 Seek time, 561
 Segmentation, 481
 Selector values, C-10
 Semiconductors, 45
 Send message routine, 627
 Sensitivity list, C-24
 Sequencers
 explicit, D-32
 implementing next-state function
 with, D-22–28
 Sequential logic, C-5
 Servers
 cost and capability, 5
 defined, 5
 See also Desktop and server RISCs
 Set-associative caches, 465–66
 address portions, 470
 block replacement strategies, 507
 choice of, 506
 defined, 465
 four-way, 467, 472
 memory-block location, 466
 misses, 468–69
 n-way, 465
 two-way, 467
 See also Caches
 Setup time, C-53, C-54
 Shaders, CDA.11:3
 defined, A-14
 floating-point arithmetic, A-14
 graphics, A-14–15
 pixel example, A-15–17
 Shading languages, A-14
 Shared memory
 caching in, A-58–60
 CUDA, A-58
 defined, A-21
 as low-latency memory, A-21
 N-body and, A-67–68
 per-CTA, A-39
 SRAM banks, A-40
 See also Memory
 Shared memory multiprocessors (SMP),
 624–26
 defined, 619, 624
 single physical address
 space, 624
 synchronization, 625
 Shift operations, B1-7
 Barrel shifter circuit outputs, B1-7
 Signals

 asserted, 291, C-4
 control, 292, 306, 307, 308
 deasserted, 291, C-4
 Sign and magnitude, 233
 Sign bit, 89
 Signed division, 229–31
 Signed multiplication, 224
 Signed numbers, 86–93
 sign and magnitude, 88
 treating as unsigned, 109
 Sign extension, 296
 defined, 123
 shortcut, 91–92
 Significands, 234
 addition, 239
 multiplication, 244
 Silicon
 crystal ingot, 45
 defined, 45
 as key hardware technology, 54
 wafers, 45
 SIMD (Single Instruction Multiple Data),
 635, 645
 computers, CD7.14:1–3
 data vector, A-35
 extensions, CD7.14:3
 for loops and, CD7.14:2
 massively parallel multiprocessors,
 CD7.14:1
 small-scale, CD7.14:3
 vector architecture, 636–39
 in x86, 635–36
 SIMMs (single inline memory modules),
 CD5.13:4, CD5.13:5
 Simple programmable logic devices
 (SPLDs), C-78
 Simplicity, 172
 Simultaneous multithreading
 (SMT), 632–34
 defined, 632
 support, 633
 thread-level parallelism, 633
 unused issue slots, 634
 Single-clock-cycle pipeline
 diagrams, 342
 defined, 342
 illustrated, 344
 Single-cycle datapaths
 illustrated, 331
 instruction execution, 332
 See also Datapaths

- Single-cycle implementation
 - control function for, 313
 - defined, 313
 - nonpipelined execution versus pipelined execution, 320
 - non-use of, 314–16
 - penalty, 316
 - pipelined performance versus, 318–19
- Single-instruction multiple-thread (SIMT), A-27–30
 - defined, A-27
 - multithreaded warp scheduling, A-28
 - overhead, A-35
 - processor architecture, A-28
 - warp execution and divergence, A-29–30
- Single instruction single data (SISD), 634
- Single precision
 - binary representation, 236
 - defined, 234
 - See also* Double precision
- Single-program multiple data (SPMD), 634, A-22
- Small Computer Systems Interface (SCSI) disks, 563, 599
- Smalltalk
 - Smalltalk-80, CD2.20:7
- Snooping protocol, 522–23, 524
- Snoopy cache coherence, CD5.9:16
- Software
 - GPU driver, 641
 - layers, 10
 - multiprocessor, 618
 - parallel, 619
 - as service, 592, 672
 - systems, 10
- Sort algorithms, 152
- Sorting performance, A-54–55
- Sort procedure, 145–51
 - code for body, 146–48
 - defined, 145
 - full procedure, 149–51
 - passing parameters in, 149
 - preserving registers in, 149
 - procedure call, 148
 - register allocation for, 146
 - See also* Procedures
- South bridge, 570
- Space allocation
 - on heap, 119–22
 - on stack, 119
- Sparse matrices, A-55–58
- Sparse Matrix-Vector multiply (SpMV), 665–66, 681, A-55, A-57, A-58
 - CUDA version, A-57
 - serial code, A-57
 - shared memory version, A-59
- Spatial locality, 438–39
 - defined, 438
 - large block exploitation of, 450
 - tendency, 442
- SPEC, CD1.10:10–11
 - CPU benchmark, 48–49
 - defined, CD1.10:10
 - power benchmark, 49–50
 - SPEC89, CD1.10:10
 - SPEC92, CD1.10:11
 - SPEC95, CD1.10:11
 - SPEC2000, CD1.10:11
 - SPEC2006, 268, CD1.10:11
 - SPECPower, 583
 - SPECrate, 650
 - SPECratio, 48
- Special function units (SFUs), A-35
 - defined, A-43
 - GeForce 8800, A-50
- Speculation, 378–79
 - defined, 378
 - hardware-based, 386
 - implementation, 378
 - performance and, 379
 - problems, 379
 - recovery mechanism, 379
- Speed-up challenge, 621–24
 - balancing load, 623–24
 - bigger problem, 622–23
- Spilling registers, 85, 117
- SPLASH/SPLASH 2 (Stanford Parallel Applications for Shared Memory), 650–52
- Split caches, 456
- Stack architectures, CD2.20:3
- Stack pointers
 - adjustment, 117
 - defined, 115
 - values, 117
- Stacks
 - allocating space on, 119
 - for arguments, 151
 - defined, 114
 - pop, 115
 - push, 115, 117
- Stalls, 324–25
 - avoiding with code reordering, 324–25
 - behavioral Verilog with detection, CD4.12:5–9
 - data hazards and, 357–60
 - defined, 324
 - illustrations, CD4.12:25, CD4.12:28–30
 - insertion into pipeline, 360
 - load-use, 363
 - memory, 464
 - as solution to control hazard, 326
 - write-back scheme, 462
 - write buffer, 462
- Standby spares, 591
- State
 - in 2-bit prediction scheme, 367
 - assignment, C-70, D-27
 - bits, D-8
 - exception, saving/restoring, 501
 - logic components, 291
 - specification of, 482
- State elements
 - clock and, 292
 - combinational logic and, 292
 - defined, 291, C-48
 - inputs, 291
 - register file, C-50
 - in storing/accessing instructions, 294
- Static branch prediction, 379
- Static data
 - segment, 119
- Static multiple-issue processors, 378, 379–83
 - control hazards and, 380
 - instruction sets, 379
 - with MIPS ISA, 380–83
 - See also* Multiple issue
- Static random access memories (SRAMs), C-58–62
 - array organization, C-62
 - basic structure, C-61
 - defined, 20, C-58
 - fixed access time, C-58
 - large, C-59
 - read/write initiation, C-59
 - synchronous (SSRAMs), C-60
 - three-state buffers, C-59, C-60

- Static variables, 119
 - Status register, 576
 - illustrated, 577
 - Steady-state prediction, 366
 - Sticky bits, 255
 - Storage
 - disk, 561–65
 - flash, 566–68
 - nonvolatile, 561
 - Storage area networks (SANs),
 - CD6.11:11
 - Store buffers, 389
 - Stored program concept, 77
 - as computer principle, 99
 - illustrated, 100
 - principles, 171
 - Store instructions
 - access, A-41
 - base register, 305
 - defined, 84
 - EX stage, 339
 - ID stage, 335
 - IF stage, 335
 - instruction dependency, 357
 - MEM stage, 340
 - unit for implementing, 297
 - WB stage, 340
 - See also* Load instructions
 - Store word, 84
 - Strcpy procedure, 124–25
 - defined, 124
 - as leaf procedure, 125
 - pointers, 125
 - See also* Procedures
 - Stream benchmark, 661
 - Streaming multiprocessor (SM),
 - A-48–49
 - Streaming processors, 643, A-34
 - array (SPA), A-41, A-46
 - GeForce 8800, A-49–50
 - Streaming SIMD Extension 2 (SSE2)
 - floating-point architecture, 261–62
 - Stretch computer, CD4.15:1
 - Strings
 - defined, 124
 - in Java, 126–27
 - representation, 123
 - Striping, 587
 - Strong scaling, 623, 624
 - Structural hazards, 321–22, 338
 - Structured Query Language (SQL),
 - CD6.14:5
 - Subnormals, 257
 - Subtracks, 592
 - Subtraction, 216–21
 - binary, 216–17
 - floating-point, 247
 - negative number, 218
 - overflow, 218
 - See also* Arithmetic
 - Sum of products, C-11, C-12
 - Sun Fire x4150 server, 592–98
 - front/rear illustration, 594
 - idle and peak power, 598
 - logical connections and bandwidths, 595
 - minimum memory, 597
 - Sun UltraSPARC T2 (Niagara 2),
 - 633, 644
 - base versus fully optimized performance, 669
 - characteristics, 663
 - defined, 663
 - illustrated, 662
 - LBMHD performance, 668
 - roofline model, 664
 - SpMV performance, 681
 - Supercomputers, 5, CD4.15:1
 - Superscalars
 - defined, 383, CD4.15:4
 - dynamic pipeline scheduling, 384, 385–86
 - multithreading options, 632
 - Surfaces, A-41
 - Swap procedure, 144–45
 - body code, 144
 - defined, 144
 - full, 145
 - register allocation, 144
 - See also* Procedures
 - Swap space, 484
 - Switched networks, CD6.11:5
 - Switches, CD6.11:6–7
 - Symbol tables, 136
 - Synchronization, 133–34
 - barrier, A-18, A-20, A-34
 - defined, 625
 - lock, 133
 - overhead, reducing, 43
 - unlock, 133
 - Synchronizers
 - defined, C-76
 - from D flip-flop, C-76
 - failure, C-77
 - Synchronous bus, 569
 - Synchronous DRAM (SRAM),
 - 459, C-60, C-65
 - Synchronous SRAM (SSRAM), C-60
 - Synchronous system, C-48
 - Syntax tree, CD2.15:3
 - System Performance Evaluation
 - Cooperative. *See* SPEC
 - Systems software, 10
 - SystemVerilog
 - cache controller, CD5.9:1–9
 - cache data and tag modules, CD5.9:5
 - FSM, CD5.9:6–9
 - simple cache block diagram, CD5.9:3
 - type declarations, CD5.9:1, CD5.9:2
- ## T
- Tags
 - defined, 444
 - in locating block, 470
 - page tables and, 484
 - size of, 472–73
 - Tail call, 121
 - Task identifiers, 496
 - Task parallelism, A-24
 - TCP/IP packet format, CD6.11:4
 - Telsa PTX ISA, A-31–34
 - arithmetic instructions, A-33
 - barrier synchronization, A-34
 - GPU thread instructions, A-32
 - memory access instructions, A-33–34
 - Temporal locality, 439
 - defined, 438
 - tendency, 442
 - Temporary registers, 81, 116
 - Terabytes, 5
 - Tesla multiprocessor, 644
 - Texture memory, A-40
 - Texture/processor cluster (TPC),
 - A-47–48
 - TFLOPS multiprocessor, CD7.14:5
 - Thrashing, 503
 - Thread blocks, 645

- creation, A-23
- defined, A-19
- managing, A-30
- memory sharing, A-20
- synchronization, A-20
- Thread dispatch, 645
- Thread parallelism, A-22
- Threads
 - creation, A-23
 - CUDA, A-36
 - ISA, A-31–34
 - managing, A-30
 - memory latencies and, A-74–75
 - multiple, per body, A-68–69
 - warps, A-27
- Three Cs model, 509
- Three-state buffers, C-59, C-60
- Throughput
 - defined, 28
 - multiple issue and, 387
 - pipelining and, 330, 387
- Thumb instructions, B1-8, B1-10, B2-9
 - ADC, B1-8
 - ADD, B1-9
 - ADR, B1-10
 - AND, B1-10
 - Arithmetic shift right, B1-11
 - ASR, B1-11
 - BIC, B1-12
 - BKPT, B1-12
 - BL, B1-12
 - BLX, B1-13
 - Branch relative, B1-11
 - BXJ, B1-13
 - CDP, B1-14
 - CLZ, B1-14
 - CMN, B1-14
 - CMP, B1-15
 - CPS, B1-16
 - CPY, B1-16
 - Decode table, B1-8-10, B2-10
 - Decoding procedure, B2-9
 - Table efficiently, B2-9
 - Encodings, B2-9
 - EOR, B1-16
 - LDC, B1-17, B1-19
 - Addressing format, B1-17
 - LDM, B1-18
 - Addressing modes, B1-19
 - LDR, B1-20, B1-22
 - Addressing format, B1-20
 - Datatypes, B1-21
 - LSL, B1-22
 - LSR, B1-22
 - MCRR, B1-23
 - MLA, B1-23
 - MOV, B1-24
 - MRRC, B1-24
 - MRS, B1-25
 - MSR, B1-25
 - MUL, B1-26
 - MVN, B1-26
 - NEG, B1-27
 - NOP, B1-27
 - ORR, B1-27
 - PKH, B1-28
 - PLD, B1-28
 - POP, B1-29
 - PUSH, B1-29
 - QDADD, B1-29–30
 - REV, B1-30–31
 - RFE, B1-31
 - ROR, B1-31
 - RSB, B1-32–33
 - RSB, B1-31–32
 - S E T E N D, B1-34–35
 - SADD, B1-33
 - SBC, B1-34
 - SEL, B1-34
 - SHADD, B1-35–36
 - SMLA, B1-36–37
 - SMMLA, B1-37
 - SMUL, B1-38
 - SRS, B1-39
 - SSAT, B1-39
 - SSUB, B1-39
 - STC, B1-39
 - Addressing format, B1-40
 - STM, B1-40
 - Addressing modes, B1-41
 - Memory addresses, B1-41
 - Pseudocode, B1-40
 - Register number, B1-40
 - STR, B1-42
 - Addressing modes, B1-42, B1-43
 - Data types, B1-43
 - System coprocessor, B1-44
 - SUB, B1-44–45
 - SWI, B1-45
 - SWP, B1-45–46
 - SXT, B1-46–47
 - TEQ, B1-47
 - TST, B1-47–48
 - USAD, B1-48–49
 - See also* ARM instruction
- Timing
 - asynchronous inputs, C-76–77
 - level-sensitive, C-75–76
 - methodologies, C-72–77
 - two-phase, C-75
- TLB misses, 489
 - entry point, 500
 - handler, 500
 - handling, 496–502
 - minimization, 681
 - occurrence, 496
 - problem, 503
 - See also* Translation-lookaside buffer (TLB)
- Tomasulo's algorithm, CD4.15:2
- Tournament branch predictors, 369
- Tracks, 561
- Transaction Processing Council (TPC), 582
- Transaction processing (TP)
 - defined, 582
 - I/O benchmarks, 582–83
- Transfer time, 562
- Transistors, 26
- Translation-lookaside buffer (TLB), 488–90, CD5.13:5
 - associativities, 489
 - defined, 488
 - illustrated, 488
 - integration, 490–94
 - Intrinsity FastMATH, 490
 - typical values, 489
 - See also* TLB misses
- Tree-based parallel scan, A-62
- Truth tables, C-5
 - ALU control lines, D-5
 - for control bits, 304
 - datapath control outputs, D-17
 - datapath control signals, D-14
 - defined, 303
 - example, C-5
 - next-state output bits, D-15
 - PLA implementation, C-13
- Two-level logic, C-11–14
- Two-phase clocking, C-75

Two's complement representation, 88, 89
 advantage, 89
 defined, 88
 negation shortcut, 90–91
 rule, 92
 sign extension shortcut, 91–92
 TX-2 computer, CD7.14:3

U

Unconditional branches, 106
 Underflow, 233
 Unicode
 alphabets, 126
 defined, 126
 example alphabets, 126
 Unified GPU architecture,
 A-10–12
 illustrated, A-11
 processor array, A-11–12
 Uniform memory access (UMA), 624–25,
 A-9
 defined, 624
 multiprocessors, 625
 Units
 commit, 385, 388
 control, 289, 302–3, D-4–8, D-10,
 D-12–13
 defined, 255
 floating point, 255
 hazard detection, 358, 359
 for load/store implementation, 297
 rank, 592, 593
 special function (SFUs), A-35,
 A-43, A-50
 UNIVAC I, CD1.10:4
 UNIX, CD2.20:7, CD5.13:8–11
 AT&T, CD5.13:9
 Berkeley version (BSD), CD5.13:9
 genius, CD5.13:11
 history, CD5.13:8–11
 Unlock synchronization, 133
 Unsigned numbers, 86–93
 Use latency
 defined, 381
 one-instruction, 382

V

Vacuum tubes, 26
 Valid bit, 444

Variables
 C language, 119
 programming language, 80
 register, 80
 static, 119
 storage class, 119
 type, 119
 VAX architecture, CD2.20:3, CD5.13:6
 Vectored interrupts, 372
 Vector processors, 636–39
 conventional code comparison,
 636–37
 instructions, 638
 multimedia extensions and, 639
 scalar versus, 638
 See also Processors
 Verilog
 behavioral definition of MIPS
 ALU, C-25
 behavioral definition with bypassing,
 CD4.12:4–5
 behavioral definition with stalls for
 loads, CD4.12:6–7, CD4.12:8–9
 behavioral specification, C-21,
 CD4.12:2–3
 behavioral specification of multicycle
 MIPS design, CD4.12:11–12
 behavioral specification with
 simulation, CD4.12:1–5
 behavioral specification with stall
 detection, CD4.12:5–9
 behavioral specification with
 synthesis, CD4.12:10–16
 blocking assignment, C-24
 branch hazard logic implementation,
 CD4.12:7–9
 combinational logic, C-23–26
 datatypes, C-21–22
 defined, C-20
 forwarding implementation,
 CD4.12:3
 MIPS ALU definition in, C-35–38
 modules, C-23
 multicycle MIPS datapath, CD4.12:13
 nonblocking assignment, C-24
 operators, C-22
 program structure, C-23
 reg, C-21–22
 sensitivity list, C-24
 sequential logic specification,
 C-56–58
 structural specification, C-21
 wire, C-21–22
 Vertical microcode, D-32
 Very large-scale integrated (VLSI)
 circuits, 26
 Very Long Instruction Word (VLIW)
 defined, 379
 first generation computers,
 CD4.15:4
 processors, 380
 VHDL, C-20–21
 Video graphics array (VGA) controllers,
 A-3–4
 Virtual addresses
 causing page faults, 500
 defined, 479
 mapping from, 480
 size, 481
 Virtualizable hardware, 513
 Virtually addressed caches, 494
 Virtual machine monitors (VMMs)
 defined, 512
 implementing, 531–33
 laissez-faire attitude, 532
 page tables, 515
 in performance improvement, 514
 requirements, 513
 Virtual machines (VMs), 511–15
 benefits, 512
 illusion, 515
 instruction set architecture
 support, 513–14
 performance improvement, 514
 for protection improvement, 512
 Virtual memory, 478–503
 address translation, 479,
 488–90
 defined, 478
 integration, 490–94
 mechanism, 502
 motivations, 478–79
 page faults, 479, 484
 protection implementation,
 494–96
 segmentation, 481
 summary, 502
 virtualization of, 515
 writes, 487
 See also Pages
 Visual computing, A-3
 Volatile memory, 21

W

- Wafers, 46
 - defects, 46
 - defined, 45
 - dies, 46
 - yield, 46
- Warps, 643, A-27
- Weak scaling, 623
- Wear leveling, 567
- Web server benchmark (SPECWeb), 583
- While loops, 106–8
- Whirlwind, CD5.13:1, CD5.13:3
- Wide area networks (WANs), CD6.14:7–8
 - defined, 25
 - history of, CD6.14:7–8
 - See also* Networks
- Winchester disk, CD6.14:2–4
- Wireless LANs, CD6.11:8–10
- Words
 - accessing, 82
 - defined, 80
 - double, 165
 - load, 82, 84
 - quad, 165
 - store, 84
- Working set, 503
- Worst-case delay, 316
- Write-back caches
 - advantages, 508
 - cache coherency protocol, CD5.9:12
 - complexity, 454
 - defined, 453, 507
 - stalls, 462
 - write buffers, 454
 - See also* Caches
- Write-back stage
 - control line, 348
 - load instruction, 336
 - store instruction, 338
- Write buffers
 - defined, 453
 - stalls, 462
 - write-back cache, 454
- Write invalidate protocols, 522, 523
- Writes
 - complications, 453
 - expense, 502
 - handling, 452–54
 - memory hierarchy handling of, 507–508
 - schemes, 453
 - virtual memory, 487
 - write-back cache, 453, 454
 - write-through cache, 453, 454
- Write serialization, 521–22

- Write-stall cycles, 462
- Write-through caches
 - advantages, 508
 - defined, 453, 507
 - tag mismatch, 454
 - See also* Caches

X

- X86, 161–69
 - brief history, CD2.20:5
 - conclusion, 169
 - data addressing modes, 163, 165
 - evolution, 161–63
 - first address specifier encoding, 169
 - floating point, 259–61
 - floating-point instructions, 262
 - historical timeline, 161–63
 - instruction encoding, 168–69
 - instruction formats, 168
 - instruction set growth, 171
 - instruction types, 165
 - integer operations, 165–67
 - I/O interconnects, 570–72
 - registers, 163
 - SIMD in, 635–36
 - typical instructions/functions, 166
 - typical operations, 167
- Xerox Alto computer, CD1.10:7–8

