

INDEX

- 1-bit predictors
 - limitations, 120
 - misprediction rate, 119–20
- 2-bit predictors
 - advantages, 120
 - state diagrams, 120
- 5-stage pipelines, 92–100
 - arithmetic logic unit instructions, 93–4
 - block multi-threading in, 432–4
 - caches, 347
 - clock rates, 74
 - independent loads, 93
 - interleaved multi-threading in, 438–9
 - mechanisms, 74
 - miss latencies, 431–2
 - precise exceptions in, 99
 - prefetching, 211
 - schematics, 93, 97
 - stores, 93–4
 - translation lookaside buffers, 220
- accesses
 - atomic cache, 359–60
 - temporal order of, 366
 - see also* memory accesses
- access history, cache lines, 202
- access ordering rules, for sequential consistency, 378–9
- access times
 - disks, 11, 12
 - memories, 9, 10–11
- access types, matrices, 160
- accumulators, 80
- ACE (architecturally correct execution), 63
- active processes, 430, 431
 - blocked, 431
 - ready, 431
- activity factor, 46
- AD (area times delay) metric, 53
- Ada (programming language), 89
- adaptive networks, 350
- adaptive routing, 336–7
- ADD, 77, 80–1, 493
- ADDA, 80
- ADDI, 81, 106
- addressable registers, 81
- address generation unit (AGU), 113–14
- addressing, Big Endian vs. Little Endian, 82
- addressing modes, 82–3
 - synthesized, 83
- address translation, 274
- ADDU, 77
- Advanced Micro Devices (AMD), Hyper Transport, 455
- aging
 - components, 59
 - devices, 64
- AGU (address generation unit), 113–14
- algorithms
 - classification, 272
 - optimization, 25
 - parallel, 237–9
 - sequential, 235
 - software pipelining, 149–51
 - sorting, 25
 - thread selection, 439
 - see also* Dekker's algorithm; routing algorithms; Tomasulo algorithm
- aliases, 214
- all-path execution, 117–18
- Alpha ISA, 497–8
- Altivec, 158
- ALUs *see* arithmetic logic units (ALUs)
- AM (attraction memory), 294–5, 297
- AMAT (average memory access time), 205
- AMD (Advanced Micro Devices), Hyper Transport, 455
- Amdahl, Gene, 22
- Amdahl's law, 22–6, 79, 83, 151
 - criticisms, 25
 - and Gustafson's law compared, 26
- amperes, 38
- AND, 77
- antialiasing hardware, 223
- APIs (application-programming interfaces), 233
- application-programming interfaces (APIs), 233
- application-programming interfaces (APIs), 233
- application-specific integrated circuits (ASICs), 2
- APTs (atomic protocol transactions), 360–1, 364–5, 367
- arbitration, 318
- architecturally correct execution (ACE), 63
- architectural vulnerability factors (AVFs), 62–3
 - L1 cache bit cell analysis, 63–4
- architecture
 - chip multiprocessors, 446–59
 - innovations, 7–8
 - parallelism in, 13–17
 - see also* cache-only memory architecture (COMA); computer architecture; instruction set architectures (ISAs); microarchitectures; non-uniform memory architectures (NUMAs); very long instruction word (VLIW) architectures
- area overheads, 62
- area times delay (AD) metric, 53
- arithmetic instructions, 77
- arithmetic logic units (ALUs), 77, 481
 - instructions, pipelining, 93–4
 - memory operands, 83–4
 - power-optimized, 509, 510
- arithmetic/logic vector instructions, 159–60
- arithmetic means
 - execution times, 19
 - speedups, 20
- array processors, 15–17
 - see also* single instruction multiple data (SIMD) processors

522 **Index**

- ASICs (application-specific integrated circuits), 2
- Asim, 491
- assembly code, 4
- assist threads *see* helper threads
- asynchronous message-passing, 242
- asynchronous SEND primitives, 241–2
- asynchronous snooping, 256
- ATOM, 497–8
- atomic blocks, 465
- atomic buses, 323
- atomic cache accesses, 359–60
- atomic events, 358–9
- atomic memory accesses, 360–1
- atomic protocol transactions (APTs), 360–1, 364–5, 367
- attraction memory (AM), 294–5, 297
- automatic parallelization, with OpenMP, 462–3
- average memory access time (AMAT), 205
- average routing distance, 317
- AVFs *see* architectural vulnerability factors (AVFs)
- back-end, 113, 133
 design issues, 138
- backpointers, 223
- backward compatibility, 77
 concept of, 4
- backward slices, 479
- bandwidth
 bisection, 319
 cache, 453–5
 effective, 318
 link, 315
 off-chip, 453–4
 see also memory bandwidth
- bandwidth models, 318–19
- BAR (barrier count), 391
- barrel processors, 440–1
 examples, 441–2
- barrier count (BAR), 391
- barriers, 391–2
- barrier synchronization, 237, 504–5, 507
- baseline directory protocols
 behavior, 279–82
 memory requirements, 283–4
 reducing latencies in, 282–3
- basic block vectors (BBVs), 513–14
- bathtub curves, 59
- BBVs (basic block vectors), 513–14
- behavioral specifications
 cache protocols, 354
 state-transition diagrams as, 252–3
- benchmarking, 18–22
- benchmarks, 490
 synthetic, 515
- benchmark suites, 18–19, 489
 roles, 18
- BEQ, 153
 branch handling, 98
- Berkeley Spice Simulation, 510
- BEZ, 104, 120
- Big Endian addressing, vs. Little Endian addressing, 82
- binary code, and high-level languages, 4
- binning
 definition, 70
 and die-to-die variations, 70
- bisection bandwidth, 319
- bitarea, definition, 61
- bit overheads, 62
- blocked threads, 431
- blocking, 249, 397
- blocking caches
 definition, 207
 and non-blocking caches compared, 208–9
- blocking message-passing primitives, 242–3
- block localization, 295
- block multi-threading, 432–8
 in 5-stage pipelines, 432–4
 and cache misses, 434
 in out-of-order cores, 434–7
 processors with, 437–8
 speculative scheduling with, 436–7
- block relocation, 295
- BNEZ, 104
- bottlenecks
 elimination, 309
 performance, 514–15
- BPBs *see* branch prediction buffers (BPBs)
- branches, 78–9
 complex, 97–8
 delayed, 106
 unresolved, execution beyond, 117–18
 see also jumps
- branch handling, 98
 multiple, 138
- branch prediction, 104–5
 hardwired, 104
 static, 104
 see also dynamic branch prediction
- branch prediction buffers (BPBs), 118–19, 124
 with global branch history, 121
- branch predictors
 combining, 122, 123
 correlating, 120–1
- branch target buffers (BTBs), 122–3, 124
- breakpoints, 85
- broadcall, definition, 348–9
- broadcast, definition, 348–9
- broadcast requests, 312
- BTBs (branch target buffers), 122–3, 124
- bubble sort, 25
- buffer overflows, 269
- buffers
 branch target, 122–3, 124
 physical, 334
 real store, 399
 see also branch prediction buffers (BPBs); re-order buffers (ROBs); translation lookaside buffers (TLBs)
- burn-in, 58–60
- burn-in testing, 59
- bus-based shared-memory
 multiprocessor systems, 246–76
- buses, 322–3
 applications, 348–9
 atomic, 323
 bandwidth, capped, 276
 and cache coherence, 348–9
 input/output, 5
 interconnections, 5
 non-pipelined, 323
 pipelined, 323
 snooping, 256
 in symmetric multiprocessors, 362
 system, 349
 see also split-transaction buses
- BusRd *see* bus-read request (BusRd)
- BusRdX *see* bus-read-exclusive request (BusRdX)
- bus-read-exclusive request (BusRdX), 252, 253, 266, 281, 282–3, 286–7, 288, 357, 363, 380, 381–2, 472
 definition, 251

- bus-read request (BusRd), 253, 264, 278, 279–80, 281, 282–3, 286, 288, 380, 381–2, 472
 - definition, 251
- bus update (BusUpdate), 262–3, 264, 380
- BusUpdate (bus update), 262–3, 264, 380
- bus-upgrade request (BusUpgr), 254, 264, 266, 268, 277–8, 279, 280, 281, 282–3, 286
- BusUpgr (bus-upgrade request), 254, 264, 266, 268, 277–8, 279, 280, 281, 282–3, 286
- BusWrite *see* bus-write request (BusWrite)
- bus-write request (BusWrite), 252, 253
 - definition, 251
- busy bits, 281
- busy waiting, 397
- butterfly networks, 326
 - schematics, 325
- byte addressability, 81–2
- bytes, 87

- C++, 3, 76, 233
- cache bandwidth, chip multiprocessors, 453–5
- cache-centric directory protocols, 278, 285–7, 293
- cache coherence, 248–9
 - buses and, 348–9
 - components, 3
 - definition, 249
 - maintenance, in multi-level cache hierarchies, 269
 - multiple clusters, 288
- cache coherence problem, 248, 351–4
- cache coherence protocols, 345
 - generalized class of, 258–60
- cache-coherent non-uniform memory architectures (cc-NUMAs), 289, 296–7
- cache protocols, 354
 - correctness issues, 342
 - definition, 277
 - directory protocols, 277–8, 293–4, 356–7
 - home nodes, 281–2
 - organization, 277
 - plain coherence, 369
 - snoopy cache protocols, 277
 - store atomicity in, 363–5
- cache designs, enhancements, 193
- cached setups, 516
- cache exceptions, 155
- cache hierarchies, 5, 193, 198–211, 380, 454–5
 - applications, 10–11
 - exclusion, 197
 - inclusion, 197
 - multi-level, 348
 - cache coherence maintenance, 269
 - see also* memory hierarchies
- cache hierarchy performance, 204–5
 - metrics, 204
- cache hits, multiple, 193
- cache indexing, 199
- cache lines, 193, 223
 - access history, 202
 - fault containment, 56–7
 - latency, 62
- cache mapping, 198–201
- cache misses, 155
 - and block multi-threading, 434
 - classification, 205–7, 271–3
 - multiple, 193
- cache-miss models, 4C, 270
- cache-only memory architecture (COMA)
 - block relations, 297
 - correctness issues, 342
 - design issues, 294–6
 - hardware structures, 294–6
 - structure, 294
 - see also* flat cache-only memory architecture
- cache-only shared-memory multiprocessor systems, 293–7
 - concepts, 294–6
 - hardware structures, 294–6
 - protocols, 294–6
- cache organizations
 - multiprocessor, 246–9
 - private, 246–8
 - shared, 246–8
- cache prefetching, 209–11
 - types of, 209
- cache preloading, 209–11
- cache protocols, 354–7
 - behavioral specifications, 354
 - inputs, 252, **253**
 - outputs, 252, **253**
 - on split-transaction buses, design issues, 267–9
 - variations, 260–4
- see also* invalidation-based cache protocols; snoopy cache protocols; update-based cache protocols
- caches, 9, 196, 347–8
 - design issues, 252
 - directory, 285
 - drowsy, 53
 - dynamic voltage frequency scaling, 61
 - first-level, 223
 - fully associative, 201
 - instruction trace, 138
 - invention, 36
 - L1, 447–55, 493
 - L2, 447–55, 493
 - local, 250
 - narrow, 198–9
 - organization, 198–201
 - remote, 250
 - structure, 193
 - transaction, 469–73
 - transient faults in, 62
 - wide, 198–9
 - see also* blocking caches; direct-mapped caches; I-caches; lockup-free caches; non-blocking caches; on-chip caches; set-associative caches; virtual-address caches; write-back caches; write-through caches
- cache simulation, 206
- cache states
 - non-atomic, 266–7
 - transient, 266–7
- capacitance, 38
 - measurement, 508–9
- capacitors, 39
- capacity misses, 206
 - determination, 206
- carrier tunneling, 51
- CAS (compare_and_swap), 395, 396
- cc-NUMAs *see* cache-coherent non-uniform memory architectures (cc-NUMAs)
- CCs (condition codes), 78
- CC simulations *see* cycle-by-cycle (CC) simulations
- CDB (common data bus), 113
- CDC *see* Control Data Corporation (CDC)

- central processing units (CPUs)
 definition, 6
 multiple, 2
 superpipelined, 103–4
 superscalar, 103–4
see also cores
- chaining (vector processing), 161–3
- channel-dependence graphs, 332–4
- channels, 333
 virtual, 334–5
- charges, 37–8
- chip designs, improved, 1
- chip multiprocessors (CMPs), 6, 425–87
 advantages, 425
 applications, 425
 architecture, 446–59
 shift to, 36
 bus-based, 447–8
 cache bandwidth issues, 453–5
 characteristics, 447
 classification, 446–7
 conjoined cores, 458–9
 core multi-threading, 429–46
 correctness issues, 342
 crossbar interconnects, 450
 faults in, 56
 heterogeneous, 446–7, 455–8
 and higher performance, 4–5
 homogeneous, 446–55
 interconnection networks, 309–10
 many-core, 446–7
 memory bandwidth issues, 453–5
 multi-core, 446–7
 opportunities, 428–9
 and parallel processing, 425
 programming models, 459–82
 rationale, 426–9
 ring-based, 448–50
 shared cache architecture, 450–3
 and shared-memory multiprocessor
 systems compared, 425, 428–9
 technology trends, 426–8
 use of term, 2
see also multi-core microprocessors
- circuit design, and clock rates, 7
- circuit switching, 313–14, 319–20
- circular dependency chains, 332
- CISC *see* complex instruction set computer (CISC)
- CISC ISAs
 dealing with, 138–40
 definition, 87
 front-end for, 139
 and hardware, 90
- classification algorithms, 272
- clean misses, 258
- clock frequency *see* clock rates
- clock gating, 47
- clock per instruction (CPI), 17–18
 variance measurement, 512
- clock rates
 5-stage pipelines, 74
 dynamic pipelines, 75
 factors affecting, 6–7, 44
 higher, 1
 increased, 2
 Intel microprocessors, 7
 limitations, 8
- CMOS *see* complementary metal oxide silicon (CMOS)
- CMOS endpoint, 26, 30
 use of term, 1
- CMOS inverters *see* complementary metal oxide silicon (CMOS) inverters
- CMPs *see* chip multiprocessors (CMPs)
- coarse-grain multi-threading *see* block multi-threading
- coarse-vector directory protocols, 285
- code fragments *see* programs
- code profiling, 105
- code segments *see* programs
- coherence, 342–424
 formal model of, 365–6
 issues, 222
 in multiprocessors, issues, 351–4
 need for, 193
 privacy principle, 369
 single-thread, enforcement, 351
 and store atomicity, 350–74
see also cache coherence; memory hierarchy coherence; plain coherence; strict coherence
- coherence misses, 206, 270
- coherence orders, conflicts, 372–3
- coherence transactions, 265–6, 357
- cold misses
 definition, 205
 determination, 206
- COMA *see* cache-only memory architecture (COMA)
- common data bus (CDB), 113
- communication events, classification, 270–4
- compact thermal models (CTMs), 510
- compare_and_swap (CAS), 395, 396
- competitive snooping, 264
- compilers, 74
 design, 3
 functions, 3, 76
 optimization, 74–5
 static machines, 140
 static pipelines, 110
see also parallelizing compilers
- compiling, microcodes, 90
- complementary metal oxide silicon (CMOS), 42
see also multi-threshold complementary metal oxide silicon (MTCMOS)
- complementary metal oxide silicon (CMOS) inverters
 dynamic power dissipation, 45
 ideal behavior, 50
 principles, 42–3
 schematics, 43
- complex instruction set computer (CISC)
 definition, 87
 and hardware, 90
 vs. reduced instruction set computer, 87–91
see also CISC ISAs
- component failure rate, bathtub curves, 59
- component reliability, 58
- components
 aging, 59
 cache coherence, 3
 hardware, 346–50
 latency, 315–16
 memory systems, 9
 parallel computer architecture, 5–13
 processor microarchitecture, 74
- compulsory misses, definition, 205
- compute processors (CPs), 246
- computer architects
 goals, 3, 489
 roles, 2, 62
- computer architecture, 2–5
 definition, 2
 evolution, 36
 field of study, 1
 simulations, 488
 technological impacts, 36–73
see also parallel computer architecture

- computer simulations, numerical programs, 232
- computer systems
 design, 3
 layered view of, 3, 76
 performance improvements, 232
- concurrency control mechanisms, 466
- conditional gating, 47
- conditional statements, 163
- condition codes (CCs), 78
- conductivity, 38
- conductors, 38
- conflict detection, eager vs. lazy, 466–7
- conflict misses, 206
 determination, 206
- congestion control, 338–9
- conjoined cores, 458–9
- conservative memory models,
 enforcement, in out-of-order processors, 411–12
- consistency
 sequential, 375–88
see also memory consistency; release consistency; sequential consistency
- content-addressable memory (CAM), 201, 509
 and random-access memory compared, 201
- contention, shared resources, 193
- context switches, 430
- Control Data Corporation (CDC), 158
 CDC6600 processor, 441
- control hazards, 102
- controllers, non-blocking caches, 207
- control signals, 93
- cooperative sharing, 454
- coordination, 234
- core ISA
 applications, 86–7
 instructions, 87, **88**, 91–2
 microops, 138–9
- core multi-threading, 8, 429–46
 hardware-supported, 430–2
 software-supported, 429–30
 use of term, 431
see also block multi-threading; hardware multi-threading (HMT); interleaved multi-threading; simultaneous multi-threading (SMT)
- core performance, factors affecting, 6
- cores
 clusters, 369
 conjoined, 458–9
 definition, 6
 multiple, 6
 multi-threaded, 6, 429
see also central processing units (CPUs)
- correctness, issues, 342
- CPI *see* clock per instruction (CPI)
- CPs (compute processors), 246
- CPUs *see* central processing units (CPUs)
- Cray-1 architecture, 158
- Cray machines, 158
- critical latency, 506
- critical section synchronization, 359
- crossbar interconnects, in chip multiprocessors, 450
- crossbar switches, 323–4
- CTMs (compact thermal models), 510
- current
 definition, 38
see also leakage current
- cutthrough switching, 316
- cut-through switching
 packet switching under, 320–1
 virtual, 321–2
- cycle-accurate simulators, vs. functional simulators, 491–4
- cycle-by-cycle (CC) simulations, 507
 definition, 505
 mechanisms, 506
 single-threaded, 505
- cyclic instruction scheduling, 107
- cyclic scheduling, 140, 151
- D (dirty node), 357
- data dependencies
 enforcing, 112–16
 types of, 94–7
- data dependency graphs (DDGs), 149
- data-flow graphs, 136
- data-flow limits, 136–7
 definition, 137
- data forwarding, 74, 439
 thread-aware, 443
- data hazards, 100–1
 dealing with, 94–7
- data prefetching, 454, 478–9
- D-bit (dirty bit), 216, 254, 279, 283, 297, 453
- DBNE (delayed branch not equal), 145
- DDGs (data dependency graphs), 149
- deadlock avoidance, 332–4
- DEC (Digital Equipment Corporation), Vax-11 ISA, 89
- decisions, 140
- Dekker's algorithm, 375, 401, 405, 408
 definition, 391
 failure, 402
- delay, measures, 44–5
- delayed branches, 106
- delayed branch not equal (DBNE), 145
- delinquent loads, 479
- Denelcor, Heterogeneous Element Processor, 441–2
- depletion regions, 40
- design
 cache, 193
 low-power, 46–9
 productivity, 29
- design complexity
 microprocessors, 488
 technological issues, 29–30
- design quality, in very large scale integration, 53
- design space, of interconnection networks, 311–19
- design verification
 constraints, 29
 improvements, 29
- DESs (discrete-event simulations), 504
- destructive interference, 454
- detected but unrecoverable errors (DUEs), 56
 definition, 55
 detection, 62
 goals, 58
- deterministic routing, 332–4
- devices
 aging, 64
 characteristics, **44**
- device variations, classification, 70
- dielectrics, 39
 materials, 51
see also gate dielectrics; time-dependent dielectric breakdown (TDDB)
- die-to-die variations
 binning and, 70
 definition, 70
- Digital Equipment Corporation (DEC), Vax-11 ISA, 89
- dimension-order routing, 331
- diminishing returns, law of, 23–4

- DIMMs (double inline memory modules), 455
- direct-execution simulations, 496–8
- direct interconnection networks, 322, 326–30
- direct jumps, 79
- direct-mapped caches, 198–9
 advantages, 200
 disadvantages, 200
- direct memory access (DMA), 13, 245
 controllers, 85
- direct memory overhead, 296
- directory caches, 285
- directory inter-cluster coherence, 288
- directory intra-cluster coherence, 288–9
- directory protocols
 cache-centric, 278, 285–7, 293
 cache-coherent non-uniform memory architectures, 277–8, 293–4, 356–7
 coarse-vector, 285
 concepts, 277–8
 hierarchical systems, 289
 implementations, 278–83
 with lower memory overhead, 284–5
 limited-pointer, 284, 285
 memory-centric, 278
 scalability, 283–7
 terminology, 277–8
see also baseline directory protocols
- dirty bit (D-bit), 216, 254, 279, 283, 297, 453
- dirty misses, 258, 280
- dirty node (D), 357
- discrete-event simulations (DESSs), 504
- disk densities, growth, 12
- disk memory, 9
 and main memory, speed gaps, 193–4
 speed, 193
- disks, 11–12
 access times, 11, 12
 latency, 11
 seek times, 11
 transfer times, 11
- dispatch, 140
- DMA *see* direct memory access (DMA)
- DN, 256, 257
- DONE, 256
- double inline memory modules (DIMMs), 455
- double words, 87
- Dragon multiprocessors, update-based cache protocol, 262, 263, 264
- DRAMs *see* dynamic random-access memories (DRAMs)
- drowsy caches, 53
- dual-cache directories, 223
- dual-tag directories, 257
- DUEs *see* detected but unrecoverable errors (DUEs)
- DVFS *see* dynamic voltage frequency scaling (DVFS)
- dynamically scheduled pipelines, 111–40
 advantages, 140
 disadvantages, 140
- dynamic branch prediction, 118–23
- dynamic data structure sharing
 with locks, 464
 with transactions, 466
- dynamic instruction mixes
 composition, 79–80
 definition, 79
- dynamic memory disambiguation, 126–8
- dynamic MTCMOS, 52
- dynamic page placement, 289–90
- dynamic pipelines
 characteristics, 75
 clock rates, 75
 wide superscalar, 75
- dynamic power, 27, 45–50
 definition, 37
 reduction strategies, 37, 48–9
 and supply voltages, 37
 in very large scale integration, 39
- dynamic power equation, 46
- dynamic power relation, 48–9
- dynamic random-access memories (DRAMs), 455
 error correction, 28
 improvements, 11
 as main memory, 10
 performance issues, 10–11
- dynamic techniques, duality of, 140–1
- dynamic voltage frequency scaling (DVFS), 342, 457, 458, 510
 caches, 61
 limitations, 48
- e-cube routing, 332
- ED (energy–delay) product, 54
- EEMBC (Embedded Microprocessor Benchmark Consortium), benchmarks, 19
- effective addresses, 81, 212
- effective bandwidth, 318
- electrical noise, and transient faults, 61
- electric fields, 38
- electricity, laws of, 37–9
- electric potential, 38
- electromigration (EM), 37, 64–6
 definition, 64–5
 reversibility, 66
 static random-access memories, 65–6
 in vias, 65
- else* clause, 163
- EM *see* electromigration (EM)
- Embedded Microprocessor Benchmark Consortium (EEMBC), benchmarks, 19
- embedded systems, evaluation, 19
- endianness, configuration, 82
- end-to-end packet latency
 determination, 317, 320
 models, 314–17
 unloaded, 315
- energy, 45–54
 definition, 38, 54
 metrics, 53–4
 technological issues, 27–8
- energy–delay (ED) product, 54
- energy spent per instruction (EPI), 54, 456–7
 throttling, 54
- EPI *see* energy spent per instruction (EPI)
- EPIC (explicitly parallel instruction computing) microarchitectures, 157–8, 437
- error correction, 28
- error correction codes, applications, 62
- error detection, 28
 applications, 62
- errors
 hard, 313
 recoverable, 55
 soft, 313
 vs. faults, 55–7
see also detected but unrecoverable errors (DUEs); silent data corruption (SDC) errors
- essential misses, 271
 definition, 273
- essential traffic, definition, 274

- etching, and light wavelength, 69–70
- Euclidean distance, 513
- evaluation stacks, 80–1
- EX (execute), 92, 94, 96–7, 99, 100, 102–3
- exceptions, 74, 84–6, 155–7
 - cache, 155
 - causes, 85
 - deferred, 156
 - floating-point, 85
 - integer, 85
 - silent, 99
 - support for, 155–6
 - triggering, 82
 - types of, 85–6
 - see also* precise exceptions
- exclusion, cache hierarchies, 197
- exclusive access, 322
- execute (EX), 92, 94, 96–7, 99, 100, 102–3
- execution
 - all-path, 117–18
 - beyond unresolved branches, 117–18
 - multi-path, 118
 - under speculative instruction scheduling, 133–5
 - under speculative Tomasulo algorithm, 127–8
 - under Tomasulo algorithm, 115–16
 - see also* out-of-order (OoO) execution; speculative execution; speculative instruction execution
- execution-driven simulations, 496
- execution times
 - arithmetic means, 19
 - definition, 17
 - determination, 17–18
 - see also* latency
- explicitly parallel instruction computing (EPIC) microarchitectures, 157–8, 437
- explicit register renaming, 128–31
 - schematics, 129
- explicit thread parallelization, 461–3
- F&ADD (fetch_and_add), 395
- F&OP (fetch_and_op), 395
- failsafe, use of term, 28
- failure mechanisms, 37
- failure rate, 58–60
 - component, bathtub curves, 59
 - definition, 58
- failures-in-time (FIT) rates
 - definition, 57
 - determination, 57–8
 - goals, 58
 - see also* intrinsic failures-in-time (FIT) rates
- false sharing misses, 271
 - definition, 273
- fan-out-of-4s (FO4s)
 - definition, 44
 - schematics, 45
- fat trees, 325–6
- faults
 - in cache lines, containment, 56–7
 - categorization, 28, 55
 - effects of process variations on, 69–70
 - intermittent, 28, 64–9
 - permanent, 28, 69
 - vs. errors, 55–7
 - see also* page faults; transient faults
- F/E (full/empty) bit, 136
- FeS2, 500
- fetch_and_add (F&ADD), 395
- fetch_and_op (F&OP), 395
- FETs *see* field effect transistors (FETs)
- field effect transistors (FETs)
 - definition, 39
 - see also* metal oxide silicon field effect transistors (MOSFETs)
- FIFO instructions *see* first in first out (FIFO) instructions
- fine-grain multi-threading *see* interleaved multi-threading
- finite state machines (FSMs), 219
- first in first out (FIFO) instructions, 112–13, 124, 203, 265–6
 - buffer overflows, 269
 - circular buffers, 125
 - inbound/outbound buffers, 269
 - order, 413
 - queues, 214
- first-touch page placement, 292
- FIT rates *see* failures-in-time (FIT) rates
- FLAG, 398, 406, 408
- flash memories, 9
- flat cache-only memory architecture, 296–7
 - hardware structures, 297
 - organization, 297
- flexible routing algorithms, 338
- flits (flow-control units), 321
- floating-point data types, 87
- floating-point exceptions, 85
- floating-point (FP) instructions, 78, 100, 102
- floating-point operands, 87
 - double-precision, 87
- floating-point units (FPUs), 458–9, 481
- flow control, 314
 - link-level, 338–9
- flow-control time, 318
- flow-control units (flits), 321
- flux, definition, 61
- FO4s *see* fan-out-of-4s (FO4s)
- Fortran, 3, 76, 233
- forwarding, store-to-load relaxation with, 402–4
- forwarding store buffers, plain coherence in, 366–9
- FP (floating-point) instructions, 78, 100, 102
- FPUs (floating-point units), 458–9, 481
- free space, permittivity of, 39
- front-end, 112–13
 - for CISC ISAs, 139
- front-end register alias tables, 128–30, 137–8
- FSMs (finite state machines), 219
- FU (functional unit), 136
- full/empty (F/E) bit, 136
- full-system simulators, 494
 - vs. user-level simulators, 490–1
- fully associative caches, 201
- fully connected interconnection networks, 310
- functional-first simulation, definition, 498–9
- functional simulators, vs. cycle-accurate simulators, 491–4
- functional unit (FU), 136
- function-first integrating simulators, 489, 498–9
- function-level parallelism, 235
- function pipelines, 235
- gate delays, 44–5
- gate dielectrics
 - materials, 51
 - permittivity, 51
- gate leakage, 51
- gate length, processing issues, 69–70
- gather operations, 164
- GCTs (group completion tables), 445

528 **Index**

- GEMS (General Execution-driven Multiprocessor Simulator), 500, 502
- General Execution-driven Multiprocessor Simulator (GEMS), 500, 502
- general-purpose registers (GPRs), 445
- geometric means
 advantages, 20
 speedups, 20
- global branch history register, 121
- global coherent, 403
- global instruction scheduling, 107–10
- globally performed, 408
- global scheduling, 140, 143
- global time, simulation threads, 507
- global wires, 29
- GPRs (general-purpose registers), 445
- granularity, parallelism, 236
- group completion tables (GCTs), 445
- gshare, 121
- guardian, 154
- Gustafson's law, 25–6
 and Amdahl's law compared, 26
- H (home node), 357
- hafnium dioxide, 51
- half-words, 87
- handshake signals, 256
- hard disk drives (HDDs), memory storage, 9
- hard errors, 313
- hardware alarms, 86
- hardware-based synchronization, 392–3
- hardware components, 346–50
- hardware failures, 86
- hardware layers, and instruction sets, 4
- hardware multi-threading (HMT)
 applications, 437
 principles, 437
- hardware performance, and software requirements, 4
- hardware prefetching, 209–11
 and speculative execution, 210–11
- hardware substrates, 4
- hardware thread contexts, 6, 431
- hardware transactional memory (HTM), 467
- hardwired branch prediction, 104
- harmonic means, speedups, 20
- hazard detection units (HDUs), 97, 100–1
- hazard function, 58
see also failure rate
- hazards
 control, 102
 memory, 475–6
 structural, 99, 101–2
see also data hazards; read after write (RAW) hazards; write after read (WAR) hazards; write after write (WAW) hazards
- Hazucha–Svenson model, 61
- HDDs (hard disk drives), memory storage, 9
- HDUs (hazard detection units), 97
- head-of-line (HOL) blocking, 338
- helper threads, 478–80
 synonyms, 478
- HEP (Heterogeneous Element Processor), 441–2
- Heterogeneous Element Processor (HEP), 441–2
- HHLs *see* high-level languages (HHLs)
- hierarchical multiprocessor systems, 287–9
 organization, 287
- hierarchical page tables, 216–18
 structure, 217
 three-level, 216–17
- high-end servers, 388
- high-*k* materials, 51
- high-level languages (HHLs), 3, 76
 and binary code, 4
 statements, 84
- hi (variable), 237
- HMT *see* hardware multi-threading (HMT)
- HOL (head-of-line) blocking, 338
- home node (H), 357
- home nodes, 280–1
- homonyms, 214
- Horizon, 442
- horizontal microcodes, 90
- HotLeakage, 510
- hot sets, 200
- HotSpot, 510
- hotspots, 49–50
- HT (Hyper Transport), 455
- HTM (hardware transactional memory), 467
- HTT (Hyper Threading Technologies), 445
- hypercubes, 328–30, 332
- Hyper Threading Technologies (HTT), 445
- Hyper Transport (HT), 455
- IBM *see* International Business Machines Corporation (IBM)
- IC (instruction count), 17–18
- I-caches, 139
 blocks, fetching across, 138
- ICPP (International Conference on Parallel Processing), 1
- ID (I-decode), 92, 93, 96, 98, 99, 100–1, 111, 113, 198
- ideal replacement policy (OPT), 202, 206
- ideal speedup, 24
- I-decode (ID), 92, 93, 96, 98, 99, 100–1, 111, 113, 198
- IF (I-fetch), 92, 93, 98, 118–19, 139, 140, 443
- I-fetch (IF), 92, 93, 98, 118–19, 139, 140, 443
- IFQ *see* instruction fetch queue (IFQ)
- if-then-else statements, 151, 152
- ILP *see* instruction-level parallelism (ILP)
- immediate operands, 81
- impedance, 38
- implicit, use of term, 344
- inbound message management, 379–85
- inclusion, cache hierarchies, 197
- indirect interconnection networks, 322–6
 disadvantages, 326
- indirect jumps, 79, 98–9
- indirect memory overhead, 296
- in-flight instructions, in pipelines, 427
- information revolution, 1
- in-order (IO) processors, 347
- input operands, 77
- input/output (I/O)
 coherence issues, 351
 events, 351
- input/output (I/O) buses, 5
- input/output (I/O) device interrupts, 85
- input/output (I/O) devices, 5
- input/output (I/O) interconnects, 13
- INs *see* interconnection networks (INs)
- instruction count (IC), 17–18
- instruction fetch queue (IFQ), 112–13, 440, 445, 481
 filling, issues, 138
- instruction flushing, thread-aware, 443

- instruction-level parallelism (ILP), 8, 14–15, 48, 426–7, 461
 - and dynamically scheduled pipelines, 111
 - and loop-carried dependencies, 148
 - out-of-order processors, 75
- instruction mixes, 79–80
 - definition, 79
 - static, 79
 - see also* dynamic instruction mixes
- instruction operands, 80–4
 - addressing modes, 82–3
 - inside the CPU, 80–1
- instruction per clock (IPC), 18
- instruction renaming, 137
- instructions
 - arithmetic, 77
 - arithmetic/logic vector, 159–60
 - classification, 77–9
 - floating-point, 78, 100, 102
 - formats, 87, 89
 - in-flight, 427
 - issue, register fetch after, 130–2
 - jump-and-link, 79
 - logic, 77
 - memory-access, 78
 - memory vector, 160–1
 - pipelining, 1, 91
 - activities, 92
 - predicated, 153–4
 - re-coded, 93
 - throughput improvement, 8
 - types of, 87
 - undefined, 86
 - see also* first in first out (FIFO) instructions; read–modify–write (RMW) instructions
- instruction scheduling
 - cyclic, 107
 - global, 107–10
 - local, 105–7
 - non-cyclic, 107
 - static, 105–10
 - thread-aware, 443
 - see also* speculative instruction scheduling
- instruction set architectures (ISAs), 75–91, 492–4
 - complexity, 87–9
 - implementation, 3, 76, 90, **91**
 - legacy, 87
 - memory models and, 342–3
 - register-based, 83–4
 - roles, 4
 - use of term, 2
 - see also* CISC ISAs; core ISA; load–store instruction set architectures; RISC ISAs
- instruction sets, 74
 - and hardware layers, 4
 - and software layers, 4
- instruction trace caches, 138
- instruction tracing, 85
- insulators, 38
- integer exceptions, 85
- integer operands, 87
- integrating simulators, 498–500
 - approaches, 489
 - function-first, 489
 - timing-first, 489
- Intel Corporation
 - Core i7, 455, 458
 - Hyper Threading Technologies, 445
 - i486, 427
 - IA-64, 154, 437–8, 494, 497–8
 - iAPX432 ISA, 89
 - materials, 51
 - microprocessors
 - clock rates, 7
 - feature size scaling, 8–9
 - and Microsoft, 4
 - Montecito processor, 437–8
 - Quick Path Interconnect, 455
 - simultaneous multi-threading, 444–5
 - Turbo Boost, 458
 - x86, 74, 82, 87, 90, 138, 494, 497–8, 500
 - see also* Pentium processors
- inter-cluster interconnection networks, 287
- interconnection network functions, basic, 312–14
- interconnection networks (INs), 12–13, 309–41
 - aggregate bandwidth, 319
 - characteristics, **329**
 - design concepts, 311–14
 - design space, 311–19
 - direct, 322, 326–30
 - fully connected, 310
 - importance of, 309
 - indirect, 322–6
 - inter-cluster, 287
 - intra-cluster, 287
 - mesh, 311
 - multiprocessor systems, 309–10
 - processor nodes, 5
 - roles, 309, 311
 - routing techniques, 330–7
 - switch architecture, 337–9
 - switching strategies, 319–22
 - topologies, 322–30
 - see also* multistage interconnection networks (MINs)
- interconnects, 12–13
 - crossbar, 450
 - input/output, 13
 - inter-system, 13
 - intra-system, 13
 - on-chip, 12, 13
 - on-die, 425
 - point-to-point, 349–50
 - system, 12, 13, 348–50
 - see also* vias
- interleaved memory organization, 160–1
- interleaved multi-threading, 438–42
 - in 5-stage pipelines, 438–9
 - examples, 439–40
- intermittent faults, 28, 64–9
- International Business Machines Corporation (IBM)
 - 3033 model, 112, 158
 - addressing conventions, 82
 - Cell, 457–8
 - development, 4, 76–7
 - iSeries SStar processor, 437
 - pSeries Power 4 processor, 445, 446
 - pSeries Power 5 processor, 445–6
 - pSeries Power 7 processor, 446
 - reliability goals, 58
 - simultaneous multi-threading, 444–5
 - see also* System/360 (S/360); System/370 (S/370); System/390 (S/390)
- International Conference on Parallel Processing (ICPP), 1
- International Symposium on Computer Architecture (ISCA), 1
- International Technology Roadmap for Semiconductors (ITRS)
 - leakage current predictions, 45
 - projections, 30
- Internet, 13
 - growth, 1
- interrupts, 84–6
- inter-system interconnects, 13
- inter-thread synchronization, 343

- intra-cluster interconnection networks, 287
- intra-system interconnects, 13
- intrinsic failures-in-time (FIT) rates
 definition, 62
 limitations, 62–3
- invalidation-based cache protocols
 bandwidth, 264
 optimizations, 260–2
- invalidation request (InvRq), 280–1
- Invalid blocks, 469
- inversion layers, 40
- inverted page tables, 218
- InvRq (invalidation request), 280–1
- I/O *see* input/output (I/O)
- IO (in-order) processors, 347
- I/O (input/output) buses, 5
- I/O (input/output) device interrupts, 85
- I/O (input/output) devices, 5
- IPC (instruction per clock), 18
- ISAs *see* instruction set architectures (ISAs)
- ISCA (International Symposium on Computer Architecture), 1
- isolation, transactions, 465
- iterations
 Jacobi, 345–6
 loop-carried dependencies, 149, **150**
 loops, 109–10, 461
- ITRS *see* International Technology Roadmap for Semiconductors (ITRS)
- Jacobi iterations, shared-memory programs, 345–6
- JAL (jump-and-link) instructions, 79
- Java, 3
- JIT (just-in-time) compilation, 498
- jump-and-link (JAL) instructions, 79
- jumps, 78–9
 direct, 79
 indirect, 79, 98–9
see also branches
- just-in-time (JIT) compilation, 498
- k -ary n -cubes, 328–30
- kernels, development, 3
- killer-micro, use of term, 1
- k -means clustering, 513–14
- L1 cache bit cells, analysis,
 architectural vulnerability factors, 63–4
- L1 caches, 447–55, 493
- L1-map directory, 452
- L2 caches, 447–55, 493
- LANs *see* local area networks (LANs)
- latency, 193
 cache lines, 62
 components, 315–16
 critical, 506
 disks, 11
 memory-access, 309
 models, 314–17
 of operations, 101
 unloaded, 205
see also end-to-end packet latency; execution times
- leakage current, 27, 70
 gate, 69
 per transistor, 510
 predictions, 45
 sources, 50
 subthreshold, 50
- leakage power *see* static power
- least recently used (LRU) replacement policies, 202–3
- least significant byte (LSB), 82
- legacy instruction set architectures, 87
- libraries, 76
- lifetime, 273
 definition, 352–3
- light wavelength, etching and, 69–70
- limited-pointer directory protocols, 284, 285
- linear arrays, 326–7
- link bandwidth, 315
- link-level flow control, 338–9
- links, 311, 312
 asynchronous information transfer, 312
 synchronous information transfer, 312
- Little Endian addressing, vs. Big Endian addressing, 82
- livelocks, 336
- LIW (long instruction word) machines, 442
- LL (load locked), 395–6
- loaded latency, 205
- load–load, 400
- load locked (LL), 395–6
- load request queue (LRQ), 445–6
- loads
 backward slices, 479
 delinquent, 479
- load–store, 400
- load–store instruction set architectures, 84, 87
 vector, 158–9
- load–store order, 413
- load–store queue, 113–14
- load–store relaxation, and store forwarding, 403–4
- local area networks (LANs), 13, 310
 routing algorithms, 330
- local caches, 250
- local instruction scheduling, 105–7
- locality property, 196
- local time, simulation threads, 507
- local wires, 29
- lock, 393, 395
- lock bits, 281
- lock-free data structures, 465
- locking problem, basic, 389–91
- locks, dynamic data structure sharing with, 464
- lockup-free caches, 347–8, 369, 380
 schematics, 347
see also non-blocking caches
- logic instructions, 77
- long instruction word (LIW) machines, 442
- lookahead, 504
- loop-carried dependencies, 148–9
 iterations, 149, **150**
- loop parallelization, 474–5
- loops, 106–7
 with conditional statements, 163
 iterations, 109–10, 461
 speculative parallelization of, 476
- loop unrolling, 107
 applications, 140
 disadvantages, 107–8
 superscalar processors, 108
 in very long instruction word programs, 143, **144**
- low (variable), 237
- low-power design, techniques, 46–9
- LRQ (load request queue), 445–6
- LRU (least recently used) replacement policies, 202–3
 pseudo-, 202–3
- LSB (least significant byte), 82
- M5, 502
- machine cycle time (T_c), 17–18
- mainframe computers, multiple users, 388

- main memory, 2, 5, 9, 10–11
 - access time, vs. processor cycle time, 10, 11
 - backup storage, 194
 - and disk memory, speed gaps, 193–4
 - speed, 193
 - speed gaps, 10
 - structure, 194
 - use of term, 212
- Manhattan distance, 513
- manufacturing imprecisions, and reliability, 37
- master pointers, 297
- matrices
 - access types, 160
 - sparse, 164
- matrix multiplication, parallel implementation, 290–1
- MCs (memory controllers), 161
- ME *see* memory (ME)
- mean time to failure (MTTF), 58
 - definition, 57
 - determination, 57
- MediaBench suite, 19
- MEMBARs, 404–5, 406, 408, 414
- memories, 9–12
 - access times, 9, 10–11
 - cost, 9
 - size, 9
 - see also* disk memory; main memory; random-access memory (RAM); transactional memory (TM); virtual memory
- memory (ME), 92, 94, 102
 - stages, 347
- memory-access atomicity, sufficient conditions for, 361–5
- memory-access control, 215–16
- memory accesses
 - atomic, 360–1
 - performing, 361–2
 - faster, 384–5
 - schedules, 353
- memory-access instructions, 78
- memory-access latency, 309
- memory-access locality, 196
- memory bandwidth, 276
 - chip multiprocessors, 453–5
 - sufficiency, 309
- memory blocks, 193
- memory cells, transient faults, 60
- memory-centric directory protocols, 278
- memory consistency, 342–424
 - relaxed models, 398–410
- memory-consistency models, 86
 - adherence to, 400
 - orders, 399–400
 - relaxation of, 193
 - see also* relaxed memory-consistency models
- memory controllers (MCs), 161
- memory disambiguation
 - definition, 114
 - dynamic, 126–8
 - see also* speculative memory disambiguation
- memory events, 351–2
- memory hazards, in parallel loop codes, 475–6
- memory hierarchies, 193–231
 - goals, 9
 - pyramid of memory levels, 194–5
 - schematics, 194
 - see also* cache hierarchies
- memory hierarchy coherence, 196–7
 - definition, 197
- memory inclusion, 197
- memory interleaving, and store atomicity, 373–4
- memory latency, 276
- memory levels, pyramid of, 194–8
- memory management units (MMUs), 219, 223
- memory models, 342
 - generic depiction of, 399
 - and instruction set architectures, 342–3
 - need for, 350
- memory operands, 81–2
 - hazards on, 114
 - number of, 83–4
- memory orders, speculative violations of, 411–15
- memory organization, interleaved, 160–1
- memory overheads, 295–6
 - direct, 296
 - indirect, 296
 - total, 296
- memory pressure, 296
- memory protection violation, 85
- memory space, automatic management of, advantages, 212
- memory speculation techniques, 193
- memory systems
 - components, 9
 - design criteria, 9
 - design improvements, 8
 - physical constraints, 9
- memory vector instructions, 160–1
- memory wall
 - definition, 10, 193
 - over time, 10
- meshes, 327–8, 349–50
- mesh interconnection networks, 311
- MESI protocol, 260, 262
- message management, inbound, 379–85
- message-passing
 - asynchronous, 242
 - synchronous, 240–1
- message-passing interface (MPI), 233
- message-passing multiprocessor systems, 232, 239–46
- message-passing primitives, 240–3
 - blocking, 242–3
 - non-blocking, 242–3
 - see also* synchronous message-passing primitives
- message-passing protocols, 243–4
 - hardware support for, 244–6
- message processors (MPs), 245, 246
- message transfers, between nodes, 313
- metal oxide silicon field effect transistors (MOSFETs)
 - control, 40
 - definition, 39
 - operating regions, 42
 - principles, 39–43
 - structure, 40
 - see also* nMOS transistors; pMOS transistors
- MiBench suite, 19
- microarchitectures
 - development, 2
 - evolution, 4–5
 - explicitly parallel instruction computing, 157–8, 437
 - micro-core, 7
 - Pentium processors, 7–8
 - simulation, sampling, 511–13
 - see also* processor microarchitecture
- microcodes
 - compiling, 90
 - horizontal, 90
 - implementation, 89–90
 - limitations, 90
 - use of term, 89
 - vertical, 90

- microops, 138–9
 use of term, 138
- microprocessors
 clock rates, 7
 computing power, 1–2
 design complexity, 488
 metal layers, 29
 single-chip, power consumption
 limitations, 8
see also multi-core microprocessors
- microprocessor without interlocked
 pipeline stages instruction set
 architecture *see* MIPS ISA
- Microsoft, and Intel Corporation, 4
- microwords, 90
- migratory blocks, 262
- Migratory-Clean state, 262
- Migratory-Dirty state, 262
- MIMD (multiple instruction multiple
 data) systems, 15
- miniaturization, limits, 30
- MINs *see* multistage interconnection
 networks (MINs)
- MINT, 491, 502
- MIPS ISA
 advantages, 74
 applications, 86–7
 jump-and-link instructions, 79
 ops, 141
- misaligned memory access, 85
- miss latencies, 205
 5-stage pipelines, 431–2
- miss penalties, 205
- miss rates, 204
- miss status handling registers
 (MSHRs), 207, 209
- MIT Alewife, 285
- MMUs (memory management units),
 219, 223
- MN, 257
- Modified, 381
- modified bits, 254
- Modified copy, 360–1
- MOESI protocol, 258–9, 297
 state-transition diagram, 259
 subsets, 260
- Moore's law, 2, 5, 426, 461
 formulations, 1
- MOSFETs *see* metal oxide silicon field
 effect transistors (MOSFETs)
- MOSI protocol, 297
- most recently used (MRU) replacement
 policies, 202
- most significant byte (MSB), 82
- Motorola 68000, condition codes,
 78
- MOVA, 80
- MPI (message-passing interface), 233
- MPs (message processors), 245, 246
- MRU (most recently used) replacement
 policies, 202
- MSB (most significant byte), 82
- MSHRs (miss status handling
 registers), 207, 209
- MSI-invalidate, 354, 356–7, 363,
 381–2, 467–8, 477
 strict coherence in, 360–1
- MSI protocol, 257, 258, 270, 279,
 358
 behavior, 254–5
 definition, 254
 hardware structures, 255–7
 transient states in, 266–7
- MSI-update, 354, 356–7, 382–4
- MTCMOS *see* multi-threshold
 complementary metal oxide
 silicon (MTCMOS)
- MTTF *see* mean time to failure (MTTF)
- MULT, 77
- multicast requests, 312
- multi-core microprocessors, 6
 development, 7
see also chip multiprocessors
 (CMPs)
- Multifacet, 502
- multi-path execution, 118
- multi-phase snoopy cache protocols,
 design issues, 265
- multiple branch handling, 138
- multiple clusters, cache coherence, 288
- multiple cores, 6
- multiple instruction multiple data
 (MIMD) systems, 15
- multiple instructions per clock, 137–8
- multiple users, mainframe computers,
 388
- multiprocessor cache organizations,
 246–9
- multiprocessors
 on chips, 9
 coherence in, issues, 351–4
 synchronization, 15
 use of term, 6
see also chip multiprocessors
 (CMPs); symmetric
 multiprocessors (SMPs)
- multiprocessor simulators, 500–8
 parallel, 503–8
 sequential, 501–2
- multiprocessor systems, 232–308
 design principles, 232
 historical background, 232
 interconnection networks, 309–10
 inter-thread synchronization, 343
 message-passing, 232, 239–46
see also shared-memory
 multiprocessor systems
- multistage interconnection networks
 (MINs), 324–5
 definition, 324
 routing algorithms, 330–1
- multi-threaded cores, 6, 429
- multi-threading
 software, 388
see also block multi-threading; core
 multi-threading; hardware
 multi-threading (HMT);
 interleaved multi-threading;
 simultaneous multi-threading
 (SMT)
- multi-threshold complementary metal
 oxide silicon (MTCMOS)
 circuits, 47–8
 dynamic, 52
- MULTU, 77
- MUX, 96–7
- myN rows, 238–9
- mysum (variable), 237, 239
- NAND, 77
- narrow caches, 198–9
- NBTI *see* negative bias temperature
 instability (NBTI)
- n*-by-*n* switches, 312, 338
- negative bias temperature instability
 (NBTI), 37, 66–8
 partial recovery, 68
 recovery phase, 67–8
 stress phase, 67
- NetBurst, 445
- network contention, 318
- network diameter, concept of,
 317
- network interfaces (NIs), 5, 287
 functions, 13
- networks
 adaptive, 350
 omega, 324, 330
 on-chip, 310, 312

- see also* butterfly networks;
 interconnection networks (INs);
 local area networks (LANs);
 system area networks (SANs);
 tree networks; wide area
 networks (WANs)
- networks on a chip (NoCs), 310
 network topology, 314, 317, 325
 neutron flux metrics, 60
 neutron strikes, 60
 NIs *see* network interfaces (NIs)
- nMOS transistors
 applications, 42–3
 control, 40–1
 negative bias temperature instability,
 66–7
 principles, 40–3
 structure, 41
- N/nproc rows, 236–7, 238–9
 NoCs (networks on a chip), 310
 nodes, 276–7, 322
 home, 280–1
 message transfers between, 313
 processor, 5
 routes between, 350
 shared, 357
 technology, 6
- non-atomic cache states, 266–7
 non-blocking caches, 193, 207–9
 and blocking caches compared,
 208–9
 controllers, 207
 definition, 207
 in out-of-order processors, 208, 209
see also lockup-free caches
- non-blocking message-passing
 primitives, 242–3
- non-cyclic instruction scheduling, 107
 non-cyclic very long instruction word
 scheduling, 151–3
- non-essential misses, 271
 non-pipelined buses, 323
 non-uniform cache access (NUCA), 277
 non-uniform memory architectures
 (NUMAs), 296
see also cache-coherent non-uniform
 memory architectures
 (cc-NUMAs)
- non-volatile memory, constraints, 9
 NOOPs, 143
 NOR, 77, 256
 Normal block, 468–9
 North Bridge chip, 5
- notifications, 345, 384
 nproc threads, 236–7
 NUCA (non-uniform cache access),
 277
 NUMAs *see* non-uniform memory
 architectures (NUMAs)
- numerical programs, computer
 simulations, 232
- object code, 3
 object-oriented programming, 89
 OCNs (on-chip networks), 310, 312
 off-chip bandwidth, 453–4
 Ohm's law, 38
 omega networks, 324, 330
 on-chip caches
 sleepy vs. non-sleepy mode, 52–3
 static power, 52–3
 on-chip interconnects, 12, 13
 on-chip networks (OCNs), 310, 312
 on-chip transistors, applications, 29
 on-die interconnects, topologies, 425
 on-die parallelism, 425
 OoO (out-of-order) instruction
 completion, 100–3
 OoO cores *see* out-of-order (OoO)
 cores
 OoO processors *see* out-of-order (OoO)
 processors
- Opal, 500, 502
 opcodes, 77–9
 OpenMP, 461–2, 463, 473
 automatic parallelization with, 462–3
 operand forwarding, 96–7
 operands
 alignment, 81–2
 immediate, 81
 input, 77
 integer, 87
 output, 77
 register, 100
 types of, 87
see also floating-point operands;
 instruction operands; memory
 operands
- operating system (OS), functions, 3, 76
 operating system calls, 85
 operations, latency of, 101
 ops, 143
 use of term, 141
see also microops
- op slot conflicts, 147–8
 Op-Sync order, 409
- OPT (optimum replacement policy),
 202, 206
- optimization
 algorithms, 25
 and speedups, 23
- optimum replacement policy (OPT),
 202, 206
- OR, 77
- orders
 in release consistency, 409
 sequential, 400–1
 in weak ordering, 408
- OS (operating system), functions, 3, 76
 out-of-order (OoO) architectures
 dynamic, 140
 limitations, 132
- out-of-order (OoO) cores, block
 multi-threading in, 434–7
- out-of-order (OoO) execution
 completion, 100–3
 dynamic scheduling, 111, 112
 and Tomasulo algorithm, 112
- out-of-order (OoO) processors
 conservative memory model
 enforcement in, 411–12
 dynamically scheduled, 193
 instruction-level parallelism, 75
 miss penalties, 205
 non-blocking caches, 208, 209
 performance costs, 219
 plain coherence violations, 371
 simultaneous multi-threading in,
 442–6
 speculative, 411
 value prediction, with speculative
 scheduling, 137
- output operands, 77
 overlays, 212
 ownership, concept of, 259
- packet envelopes, 313
 packets, 246
 anatomy of, 313
 routing, 313
- packet switching, 314
 under cut-through switching, 320–1
 under store-and-forward switching,
 320–1
- page coloring, 223
 page-fault handlers, 214
 page faults, 85, 194, 214, 274, 275
 page hits, 194
 page-migration schemes, 289–93

- page placement
 dynamic, 289–90
 first-touch, 292
 round-robin, 291
 static, 289–90, 291–2
- page-replication schemes, 289–93
- pages, 194
 superpages, 217
 virtual, 274
- page-table base addresses (PTBAs), 215, 216
- page-table entries (PTEs), 215, 275–6
 structure, 274–5
- page-table fragmentation problem, 216
- page tables, 215, 274
 inverted, 218
see also hierarchical page tables
- paired eviction, 223
- Palo Alto Research Center (PARC), 262
- parallel algorithms, 237–9
- parallel computer architecture
 components, 5–13
 generic, 5, 6
 motivations for, 232
- parallel computers
 historical background, 1–2
 overview, 1–35
 technological issues, 26–30
- parallel discrete-event simulations (PDESs), 504
- parallelism, 7–8, 234
 in architecture, 13–17
 function-level, 235
 granularity, 236
 on-die, 425
 single-program-multiple-date, 234
 task-level, 235
 in very long instruction word architectures, 153
see also instruction-level parallelism (ILP); thread-level parallelism (TLP)
- parallelization, 504
 automatic, 462–3
 loop, 474–5
 speculative, of loops, 476
- parallelizing compilers
 definition, 15
 limitations, 15
- parallel loop codes, memory hazards in, 475–6
- parallel multiprocessor simulators, 503–8
- parallel processing
 and chip multiprocessors, 425
 dynamic power reduction, 49
- parallel programming, development, 4–5
- parallel-programming models
 abstractions, 233–9
 definition, 233
- parallel simulations, 501
- parallel software, development, 4–5
- parallel speedups, 24–5
- PARC (Palo Alto Research Center), 262
- parity, area overhead, 62
- partial store order (PSO), 404
- partitioning, 234
- PAs (physical addresses), 215, 222, 223
- PCI (peripheral component interconnect), 5
- PC (program counter), 78, 93
- PCs *see* personal computers (PCs)
- PDESs (parallel discrete-event simulations), 504
- Pentium processors
 microarchitectures, 7–8
 Pentium 3, 138
 Pentium 4, 138, 427–8
 NetBurst, 445
 Pentium-Pro, 427
- perfect-shuffle exchanges, 324
- performance, 17–26
 hardware, 4
 improved, 232
 measures, 17
 scalar, 427–8
 technological impacts, 36
see also cache hierarchy performance
- performance bottlenecks, 514–15
- performing memory accesses, 361–2
 faster, 384–5
- performing a store, faster, 384–5
- peripheral component interconnect (PCI), 5
- permanent faults, 28, 69
- permittivity
 of free space, 39
 gate dielectrics, 51
- personal computers (PCs)
 architecture, 5
 schematics, 5
- PEs (processing elements), multiple, 16
- Peterson's algorithm *see* Dekker's algorithm
- PFEs (prefetch engines), 210
- PFV *see* presence-flag vectors (PFVs)
- PHARMSim, 494
- phits (physical transfer units), 315
- physical addresses (PAs), 215, 222, 223
- physical bits, 220
- physical buffers, 334
- physical memory, 274
- physical page frames, 194
- physical tags, virtual-address caches
 with, 221–3
- physical transfer units (phits), 315
- PIDs (process IDs), 214, 219, 237
- PIN, 497–8
- pipelined buses, 323
- pipeline depth, and clock rates, 6–7
- pipeline registers, 93
- pipelines
 deep, 7–8
 deeper, 1, 29, 427
 elongated, 139–40
 freezing, 155
 function, 235
 in-flight instructions, 427
 limitations, 8
 statistically scheduled, 91–111
 store, 369–70, 399
see also 5-stage pipelines;
 dynamically scheduled pipelines; dynamic pipelines;
 static pipelines; transmission pipelines
- pipeline stage flushing, 74
- pipelining
 applications, 91
 dynamic power reduction, 49
 instructions, 1, 91
 activities, 92
 reduced instruction set computer, 91–111
 wire delays, 29
see also software pipelining
- plain coherence, 365–73
 advantages, 371–2, 375
 definition, 366
 execution, 373
 in forwarding store buffers, 366–9
 generalizations, 369–70
 importance of, 370–2
 issues, 372–3
 use of term, 365
- pMOS transistors
 applications, 42–3
 characteristics, 42

- control, 42
 negative bias temperature instability, 66–8
- pointer overflow problem, 284–5
 pointer overflows, 284–5
 point-to-point communication, 375
 point-to-point interconnects, 349–50
 point-to-point synchronization, 392
 Poisson processes, 59
 POP, 81
 power, 45–54
 definition, 38
 growth, containment, 36
 metrics, 53–4
 and scalar performance, 427–8
 subthreshold leakage, 50
 technological impacts, 36
 on scaling, 45
 technological issues, 27–8
 types of, 45
see also dynamic power; static power
- power consumption, limitations, 8, 310
 power dissipation, 53–4
 power failures, 86
 power gating, 47, 51
 PowerPC, 158
 architecture, 218
 PowerPC processing elements (PPEs), 457–8
 power simulations, 508–10
 tools, 508
 PPEs (PowerPC processing elements), 457–8
 precise exceptions, 86, 102–3, 110–11
 dealing with, 99
 predicated instructions, 153–4
 predictors, 119–20
 combining, 122
see also 1-bit predictors; 2-bit predictors; branch predictors; two-level predictors
- PREF, 211
 prefetch degree, 209
 prefetch engines (PFEs), 210
 prefetching
 5-stage pipelines, 211
 data, 454, 478–9
 sequential, 209
 software, 211
 use of term, 209
see also cache prefetching; hardware prefetching
- preloading
 cache, 209–11
 use of term, 209
- presence-flag vectors (PFVs), 285, 297
 hardware structures for, 279
 protocols, 278–9, 281, 287
- primary misses, 207–8
- primitives
 asynchronous SEND, 241–2
 synchronous RECV, 241–2
see also message-passing primitives
- privacy principle, of coherence, 369
- private cache organizations, 246–8
- probabilities, transient faults, 62
- probe functions, 242–3
- process control blocks, 430
- processes
 definition, 6
 running, 430
 synchronization, 388
 use of term, 234
 virtual space of, 213
see also active processes; threads
- process IDs (PIDs), 214, 219, 237
- processing elements (PEs), multiple, 16
- processor cycle time, vs. main memory access time, 10, 11
- process order, use of term, 74
- processor microarchitecture, 74–192
 components, 74
 overview, 74–5
- processor nodes, interconnection networks, 5
- processor read (PrRd), 252–3
- processor reliability, 36
- processors, 5, 6–9, 347
 array, 15–17
 with block multi-threading, 437–8
 clock rates, 6–7
 compute, 246
 mass production, 6
 message, 245, 246
 scalar, 13–14
 speed, 193
 superpipelined, 74, 103–4
see also barrel processors; central processing units (CPUs); microprocessors; multiprocessors; out-of-order (OoO) processors; Pentium processors; single instruction multiple data (SIMD) processors; superscalar processors; vector processors
- processor write (PrWr), 253
- process throughput, 17
- process variations
 effects on faults, 69–70
see also die-to-die variations; within-die variations
- producer/consumer synchronization, 392
- productivity, factors affecting, 488
- program counter (PC), 78, 93
- programming
 parallel, 4–5
 with transactions, 465–6
- programming models
 chip multiprocessors, 459–82
 independent processes, 460
see also parallel-programming models
- programs
 definition, 6
 multi-threaded, 14, 15
 numerical, 232
 set of, reporting performance for, 19
 single-threaded, 14, 15
 use of term, 74
 very long instruction word, 143, 144
see also shared-memory programs
- propagation delays, 28
- protection levels, selection criteria, 62
- protocols
 MESI, 260, 262
 MOSI, 297
 presence-flag vector, 278–9
see also cache protocols; directory protocols; message-passing protocols; MOESI protocol; MSI protocol
- PrRd (processor read), 252–3
- PrWr (processor write), 253
- pseudo-least recently used replacement policies, 202–3
- PSO (partial store order), 404
- PTBAs (page-table base addresses), 215, 216
- PTEs (page-table entries), 215
- Pthreads, 461, 463, 473
- p-type bodies, 66–7
- PUSH, 80
- p-wells, 66–7
- pyramid of memory levels, 194–8
- QPI (Quick Path Interconnect), 455
- quanta, 505

- quantitative evaluations, 488–520
 quantum simulations, 505–7
 Quick Path Interconnect (QPI), 455
 quicksort, 25
- R (requester node), 357
 race conditions, 465
 random-access memory (RAM)
 and content-addressable memory
 compared, 201
 see also dynamic random-access
 memories (DRAMs); static
 random-access memories
 (SRAMs)
 random replacement policies, 202
 random sampling, 512
 systematic, 512
 random within-die variations, 70
 RATs *see* register alias tables (RATs)
 RAW hazards *see* read after write
 (RAW) hazards
 R-bit (reference bit), 216
 read after write (RAW) hazards, 95, 96,
 97, 112, 154, 475–7
 checking, 126–7
 definition, 94
 loop-carried dependencies, 148–9
 on memory operands, 114
 solving, 141
 stalling, 101
 read misses, 262
 read-miss request, 250–1
 read-modify-write (RMW)
 instructions, 395, 408
 accesses, 412, 414–15
 speculative execution of, 414
 atomicity, 391, 393–4
 definition, 393
 global access, 408
 primitives, 396
 read-write-execute (RWX) bits,
 215–16, 224
 real store buffers, 399
 receiver overhead, 316
 re-coded instructions, 93
 recoverable errors (REs), definition, 55
 RECV, 239, 240, 243–4
 reduced instruction set computer
 (RISC)
 definition, 87
 and hardware, 90
 microcodes, 90
 pipelining, 91–111
 vs. complex instruction set computer,
 87–91
 see also RISC ISAs
 redundant execution, to improve
 reliability, 480–2
 reference bit (R-bit), 216
 register alias tables (RATs), 124–6, 445
 front-end, 128–30, 137–8
 retirement, 128–30
 use of term, 128
 register-based instruction set
 architectures, 83–4
 register fetch, after instruction issue,
 130–2
 register files, 87
 structural hazards, 102
 register fills, 81
 register forwarding, 95
 register operands, forwarding paths, 100
 register renaming
 in Tomasulo algorithm, 114–15
 see also explicit register renaming
 registers, 81, 87
 addressable, 81
 general-purpose, 445
 miss status handling, 207, 209
 pipeline, 93
 poisoned, 156
 vector length, 159
 see also rotating registers
 register spills, 81
 register tags
 invalid, 113
 not_ready, 113
 ready, 113
 register transfer language (RTL), 29
 relative addresses, 331
 relaxed memory-consistency models,
 398–410
 not relying on synchronization,
 398–405
 relying on synchronization, 405–10
 relaxed memory order (RMO), Sun
 Microsystems, 404–5
 release consistency, 409–10
 orders in, 409
 speculative violations of, 415
 reliability, 54–70
 component, 58
 improvement, redundant execution,
 480–2
 manufacturing issues, 37
 metrics, 57–8
 processor, 36
 system, criticality, 56
 technological impacts, 36
 technological issues, 28
 RemAck, 283
 remote caches, 250
 remote-read request (RemRd), 280,
 282, 283
 RemRd (remote-read request), 280,
 282, 283
 re-order buffers (ROBs), 124–6, 411,
 445
 and dynamic memory
 disambiguation, 126–8
 organization, 125
 and register renaming, 128
 repeaters, 29
 replacement misses, 270, 271
 replacement policies, 201–3, 206
 definition, 202
 least recently used, 202–3
 most recently used, 202
 optimum, 201, 206
 pseudo-least recently used, 202–3
 random, 202
 replay queues, 136
 requester node (R), 357
 request phases, 265–6, 267–8
 request tables, 268
 request to send, 244
 resistance, 38
 wires, 28
 resistivity, 38
 resistors, 38
 response phases, 265–6, 267–8
 response times *see* execution times
 REs (recoverable errors), definition,
 55
 retirement register alias tables, 128–30
 Rice Parallel Processing Testbed
 (RPPT), 502
 rings, 326–7, 349–50
 RISC *see* reduced instruction set
 computer (RISC)
 RISC ISAs, 138–9
 definition, 87
 and hardware, 90
 RMO (relaxed memory order), Sun
 Microsystems, 404–5
 RMW instructions *see*
 read-modify-write (RMW)
 instructions
 ROBs *see* re-order buffers (ROBs)

- rotating register base (RRB) register, 146
- rotating registers, 149
 - solving write after read hazards with, 146–7
- round-robin page placement, 291
- routes, between nodes, 350
- routing algorithms, 314, 330–2
 - deterministic, 336
 - flexible, 338
 - simple, 330
 - west-first, 335
- routing distance, 317
 - average, 317
- routing restrictions, relaxing, 334–5, 336–7
- routing techniques, interconnection networks, 330–7
- routing time, 315
 - determination, 320
- RPPT (Rice Parallel Processing Testbed), 502
- RRB (rotating register base) register, 146
- RSIM, 491, 502
- RTL (register transfer language), 29
- Ruby, 500, 502
- running processes, 430
- running threads, 431
- RWX (read–write–execute) bits, 215–16, 224

- S/360 *see* System/360 (S/360)
- S/370 *see* System/370 (S/370)
- S/390 *see* System/390 (S/390)
- S (shared nodes), 357
- SA (speculation_active) bit, 476–7
- sampling
 - techniques, 489
 - workload, 510–14
 - see also* random sampling
- sampling microarchitecture simulations (SMARTS), 511–13
- SANs *see* system area networks (SANs)
- saturating counters, 120
- SC (store conditional), 395–6
- scalable coherent interface (SCI), 286, 287
- scalable processor architecture *see* SPARC
- scalable shared-memory multiprocessor systems, 276–93
- scalar performance, power and, 427–8
- scalar processors, 13–14
- scaled setups, 515–16
- scaling
 - advantages, 36–7
 - technology, 43–5
 - voltage, 47
 - see also* dynamic voltage frequency scaling (DVFS)
- scatter operations, 164
- scheduling, 140
 - cyclic, 140, 151
 - global, 140, 143
 - see also* instruction scheduling; speculative scheduling; trace scheduling
- SCI (scalable coherent interface), 286, 287
- SDC errors *see* silent data corruption (SDC) errors
- SECEDED code *see* single error correcting and double error detecting (SECEDED) code
- secondary misses, 207–8
- seek times, disks, 11
- semaphores, 396–7
- semiconductor devices, performance improvements, 1
- semiconductor integration, growth, 1
- semiconductors, 38
- SEND, 239, 240, 243–4
- sender overhead, 315
- sentinels, use of term, 157
- sequential algorithms, 235
- sequential consistency, 375–88, 398, 400–1
 - access ordering rules for, 378–9
 - definition, 375–7
 - difference with, 401–2
 - formal model, 376–8
 - speculative violations of, 413
 - sufficient conditions for, 378
 - violations, 377–8
- sequential multiprocessor simulators, 501–2
- sequential orders, 400–1
- sequential prefetching, 209
- sequential semantics, 234
- sequential simulations, 501
- serialization points, stores, 370
- SERs *see* single event upset rates (SERs)
- servers, high-end, 388
- SESC, 502

- set-associative caches, 200
 - N-way, 200
 - three-way, 200
- SEUs *see* single event upsets (SEUs)
- SF (speculation_fail) bit, 476–7
- SGI Challenge, 268, 269
- SGI Origin 2000, 287–8
- Shade, 491
- Shared, 259, 264, 281, 381, 472
- shared cache organizations, 246–8
 - cooperative sharing, 454
 - destructive interference, 454
- shared-memory address spaces, 276
- shared-memory communication, 344
 - models, 344–6
- shared-memory multiprocessor systems, 232, 235–8, 342
 - bus-based, 246–76
 - cache-only, 293–7
 - and chip multiprocessors compared, 425, 428–9
 - scalable, 276–93
- shared-memory multi-threaded programs, correctness issues, 342
- shared-memory programs, 236
 - Jacobi iterations, 345–6
- shared nodes (S), 357
- shared resources, contention, 193
- short misses, 222
- silent data corruption (SDC) errors, 56
 - definition, 55
 - detection, 62
 - goals, 58
- sim-cache, 494
- SIMD processors *see* single instruction multiple data (SIMD) processors
- sim-fast, 494
- Simics, 494, 499, 500
- SimOS, 494
- sim-outorder, 493–4
- SimpleMP, 494
- SimplePower, 491
- SimpleScalar, 491, 492–4, 499
- simple snoopy cache protocols, 249–53
 - behavior, 250–2
 - hardware structures for, 250
- SimPoint, 513–14
- Simpoints, 19
- simulations
 - accelerating, 488
 - cache, 206

- simulations (*cont.*)
 in computer architecture, 488
 direct-execution, 496–8
 discrete-event, 504
 execution-driven, 496
 functional-first, 498–9
 microarchitectures, sampling,
 511–13
 motivations, 489
 parallel, 501
 power, 508–10
 quantum, 505–7
 sequential, 501
 slack, 507–8
 thermal, 508–10
 timing-first, 499
see also cycle-by-cycle (CC)
 simulations; trace-driven
 simulations
- simulation slowdown, 488, 489
- simulation threads
 global time, 507
 local time, 507
- simulators
 functional vs. cycle-accurate, 491–4
 multiprocessor, 500–8
 multi-threaded, 489
 single-threaded, 489
 taxonomy, 489, 490–8
 user-level vs. full-system, 490–1
see also full-system simulators;
 integrating simulators
- simultaneous multi-threading (SMT),
 502
 definition, 442–3
 in out-of-order processors, 442–6
 with spatial redundancy, 481
 speculative scheduling with, 444
- SimWattch, 499
- single dies, thread-level parallelism,
 425
- single error correcting and double error
 detecting (SECCDED) code, 56
 applications, 62
 area overhead, 62
 definition, 55
- single event upset rates (SERs)
 definition, 61
 determination, 61
- single event upsets (SEUs)
 definition, 60
 impacts, 62–3
 mechanisms, 60–1
- single instruction multiple data (SIMD)
 processors, 458
 array processors, 15–17
 definition, 16
- single-program-multiple-data (SPMD)
 parallelism, definition, 234
- single-thread coherence, enforcement,
 351
- single-threaded simulators, 489
- slack simulations, 507–8
- slot restriction, software pipelining
 with, 147, **148**
- SMARTS (sampling microarchitecture
 simulations), 511–13
- SMPs *see* symmetric multiprocessors
 (SMPs)
- SMT *see* simultaneous multi-threading
 (SMT)
- snooping
 asynchronous, 256
 competitive, 264
 synchronous, 256
- snooping buses, structure, 256
- snoop results, reporting, 268–9
- snoopy cache protocols, 252, 288,
 354–6
 in cache-coherent non-uniform
 memory architectures, 277
 definition, 250
 design space of, 253–60
 multi-phase, design issues, 265
see also simple snoopy cache
 protocols
- snoopy inter-cluster coherence, 288
- snoopy intra-cluster coherence, 288
- soft errors, 313
- software-based synchronization, 393–8
- software layers, and instruction sets, 4
- software multi-threading, 388
- software pipelining, 107, 109–10,
 144–51
 algorithms, 149–51
 applications, 140
 with slot restriction, 147, **148**
- software prefetching, 211
- software requirements, and hardware
 performance, 4
- software transactional memory (STM),
 467
- SoftWatt, 494
- solid-state disks (SSDs), memory
 storage, 9
- sorting algorithms, 25
- source code, 3
- South Bridge chip, 5
- SPARC, 82, 494
 conjoined cores, 459
 instruction set architectures, 90, 403,
 500
 interleaved multi-threading, 439,
 440
 Niagara, 289, 446
 simultaneous multi-threading, 446
- sparse matrices, 164
- spatial locality, 196
- spatial redundancy, simultaneous
 multi-threading with, 481
- SPEC (Standard Performance
 Evaluation Corporation),
 benchmarks, 18, 516
- speculation, adding to Tomasulo
 algorithm, 123–6
- speculation_active (SA) bit, 476–7
- speculation_fail (SF) bit, 476–7
- speculative broadcast schedulers,
 135
- speculative execution
 and hardware prefetching, 210–11
 of read-modify-write instructions,
 414
- speculative instruction execution,
 117–18
 as tree of basic blocks, 117–18
 use of term, 117
- speculative instruction scheduling,
 133–6
 execution under, 133–5
 in out-of-order processors, 137
- speculative memory data management
 mechanisms, 466
- speculative memory disambiguation,
 154
 with guardian, 154
- speculative schedulers, 136
- speculative scheduling
 with block multi-threading, 436–7
 with simultaneous multi-threading,
 444
- speculative Tomasulo algorithm, 131–2
 execution under, 127–8
- speculative violations
 of memory orders, 411–15
 of release consistency, 415
 of sequential consistency, 413
 of total store order, 413–14
 of weak ordering, 414–15

- speedups
 Amdahl's law, 22–6
 arithmetic means, 20
 averaging, 20–1
 definition, 19–20
 issues, 25
 determination, 22
 geometric means, 20
 harmonic means, 20
 ideal, 24
 limitations, 22
 and optimization, 23
 parallel, 24–5
 reporting, 19–22
 superlinear, 25
- SPEs (synergistic processing elements), 457–8
- split-transaction buses, 323
 cache protocols on, design issues, 267–9
- SPMD (single-program-multiple-data)
 parallelism, definition, 234
- Squash, 477
- SRAMs *see* static random-access memories (SRAMs)
- SRQ (store request queue), 445–6
- SSDs (solid-state disks), memory storage, 9
- stage flushing, 439
 thread-aware, 443
- stale data, 271
- stalling, 74
- Standard Performance Evaluation Corporation (SPEC), benchmarks, 18, 516
- Stanford DASH system, 282
- Star machines, 158
- state bits, 215–16, 250
- state E, 259–60
- state-transition diagrams
 as behavioral specifications, 252–3
 MOESI protocol, 259
 update-based cache protocols, 263
- static branch prediction, 104
- static instruction mixes, definition, 79
- static instruction scheduling, 105–10
- static machines, compilers, 140
- static page placement, 289–90, 291–2
- static pipelines
 advantages, 110–11
 characteristics, 74
 compilers, 110
 disadvantages, 110–11
 efficiency, 74–5
 use of term, 105
- static power, 27–8, 45, 50–3
 definition, 37
 modeling, 510
 on-chip caches, 52–3
 reduction, 51–2
- static random-access memories (SRAMs), 455, 508–9
 electromigration, 65–6
 error correction, 28
 single event upsets, 62–3
 transient faults, 61
- static techniques, duality of, 140–1
- statistically scheduled pipelines *see* static pipelines
- STM (software transactional memory), 467
- storage structures, 29
- store-and-forward switching, 315–16
 packet switching under, 320–1
- store atomicity, 357–65
 advantages, 375
 in bus-based systems, 359–61
 in cache-coherent non-uniform memory architectures, 363–5
 coherence and, 350–74
 definition, 359
 and memory interleaving, 373–4
 necessary conditions for, 386
 and store synchronization compared, 387–8
 sufficient conditions for, 362, 386
 use of term, 365
- store atomic systems, 362
- store buffers, real, 399
- store conditional (SC), 395–6
- store forwarding, and load-store relaxation, 403–4
- store-load, 400
- store-load order, 413
- store pipelines, 369–70, 399
- store request queue (SRQ), 445–6
- stores, serialization points, 370
- store-store, 400
- store-store order, 413
- store synchronization, 385–8
 definition, 385–6
 and store atomicity compared, 387–8
- store-to-load relaxation
 with forwarding, 402–4
 without forwarding, 401–2
- strict coherence, 365
 definition, 358
 enforcement, 358–9
 in MSI-invalidate, 360–1
- strong ordering, 407
- structural hazards, 99, 101–2
- S/U (supervisor/user) bits, 216
- SUB, 77
- subordinate threads *see* helper threads
- subroutines, 79
- subthreshold leakage, 50
- SUBU, 77
- sum (variable), 237
- Sun Microsystems, 82, 403
 relaxed memory order, 404–5
 WildFire, 287–8
see also SPARC
- supercomputers, use of term, 158
- superlinear speedups, 25
- superpages, 217
- superpipelined processors, 74, 103–4
- superscalar processors, 103–4
 loop unrolling, 108
- superset bits, 222
- supersets, 222
- supervisor/user (S/U) bits, 216
- supply voltages
 and dynamic power, 37
 lower limits, 36
 reductions, 427
- swap, 395
- switch architecture, interconnection networks, 337–9
- switch degree, 312
- switches, 311
 4-by-4, 337
 context, 430
 crossbar, 323–4
n-by-*n*, 312, 338
- switching
 circuit, 313–14, 319–20
 cutthrough, 316
 thread, 437
 virtual cut-through, 321–2
 wormhole, 322, 334
see also cut-through switching;
 packet switching;
 store-and-forward switching
- switching elements, 312
- switching strategies, 313
 interconnection networks, 319–22
- switching time, 315–16

- symmetric multiprocessors (SMPs)
 buses, 362
 correctness issues, 342
- synchronization, 234, 388–98
 barrier, 237, 504–5, 507
 basic primitives, 389–92
 critical section, 359
 events, components, 397–8
 hardware-based, 392–3
 inter-thread, 343
 multiprocessors, 15
 point-to-point, 392
 processes, 388
 producer/consumer, 392
 in relaxed memory-consistency
 models, 405–10
 software-based, 393–8
 threads, 388
see also store synchronization
- synchronization variables, 409
- synchronous message-passing, 240–1
- synchronous message-passing
 primitives, 240–1
 advantages, 241
 disadvantages, 241
- synchronous RECV primitives, 241–2
- synchronous snooping, 256
- Sync–Op order, 409
- Sync–Sync order, 409
- synergistic processing elements (SPEs),
 457–8
- synonym problem, 215
- synonyms, 214–15, 222
 detection, 223
- synthetic benchmarks, 515
- System/360 (S/360), 22
 backward compatibility, 4, 76–7
 development, 4, 90
- System/370 (S/370), 4
 addressing conventions, 82
 development, 90
 instruction set architecture, 87
- System/390 (S/390), 4
 development, 90
- system area networks (SANs), 13, 312
 use of term, 310
- systematic random sampling, 512
- systematic within-die variations, 70
- system buses, structure, 349
- system functions
 implementation, 3–4
 on-chip, 6
- system interconnects, 12, 13, 348–50
- system reliability, criticality, 56
- system throughput, definition, 17
- System z, 4
 development, 90
- T&S (test_and_set), 393–4
- tables
 group completion, 445
 request, 268
see also page tables; register alias
 tables (RATs)
- tag checking, 199
- task-level parallelism, 235
- TBegin, 467
- TBs *see* translation lookaside buffers
 (TLBs)
- T_c (machine cycle time), 17–18
- TDDB *see* time-dependent dielectric
 breakdown (TDDB)
- technological impacts, computer
 architecture, 36–73
- technological issues, parallel
 computers, 26–30
- technology nodes, and clock rates, 6
- technology scaling, 43–5
- TEMPEST, 491
- temporal locality, 196
- TEnd, 467
- TERA, 442
- test_and_set (T&S), 393–4
- then clause, 163
- thermal simulations, 508–10
- thread communication overheads,
 impacts, 460
- thread context ID (TID), 439
- thread-level parallelism (TLP), 15, 428,
 461
 single dies, 425
 sources of, 473
- thread-level speculation (TLS), 426,
 473–8
 hardware for, 476–7
 optimizations, 477–8
- threads, 369
 blocked, 431
 definition, 6
 nproc, 236–7
 running, 431
 synchronization, 388
 use of term, 74, 234
see also helper threads; processes;
 simulation threads
- thread selection algorithms, 439
- thread sequence numbers, 476
- thread switching, 437
- threshold voltages, 40, 52
- TID (thread context ID), 439
- time-dependent dielectric breakdown
 (TDDB), 37, 68–9
 degradation, 68, 69
- time of flight, 315
- time quanta, 430
- timing-first integrating simulators, 489,
 499–500
- timing-first simulations, 499
- TLBs *see* translation lookaside buffers
 (TLBs)
- TLP *see* thread-level parallelism (TLP)
- TLS (thread-level speculation), 426,
 473–8
- TM *see* transactional memory (TM)
- Tomasulo algorithm, 112–16, 126, 140
 adding speculation to, 123–6
 advantages, 116
 execution under, 115–16
 hardware for, 112
 register renaming in, 114–15
see also speculative Tomasulo
 algorithm
- topologies, interconnection networks,
 322–30
- tori, 327–8
- total memory overhead, 296
- total store order (TSO), 403
 speculative violations of, 413–14
- TPC (Transactional Processing
 Council), benchmarks, 19
- TPC-C, 515, 516
- trace-driven simulations, 494–6
 advantages, 495
 disadvantages, 495
 variations, 496
- trace scheduling, 107, 151–2
 applications, 140
 limitations, 153
 use of term, 151
- Transactional block, 468–9
- transactional cache states, 467
- transactional conflict detection
 mechanisms, 466
- transactional memory (TM), 425–6,
 463–73
 abort, 469–73
 commit, 469–73
 hardware systems for, 467–9
 mechanisms, 466–7

- Transactional Processing Council (TPC), benchmarks, 19
- transaction caches, 469–73
- transactions
- atomic protocol, 360–1, 364–5, 367
 - coherence, 265–6, 357
 - dynamic data structure sharing with, 466
 - isolation, 465
 - programming with, 465–6
- transfer times, disks, 11
- transient cache states, 266–7
- transient faults, 27, 28, 60–4
- in caches, 62
 - causes, 60
 - electrical noise and, 61
 - probabilities, 62
 - single event upsets, 60–1
 - transistor susceptibility to, 36
- transistor densities, increased, 2, 8–9
- transistors
- on-chip, 29
 - principles, 36
 - transient faults, 36
 - see also* field effect transistors (FETs); nMOS transistors; pMOS transistors
- translation buffers (TBs) *see* translation lookaside buffers (TLBs)
- translation lookaside buffers (TLBs), 218–21, 223, 445
- 5-stage pipelines, 220
 - consistency, 274–6
 - consistency problem, 275
 - definition, 218, 274–5
 - faults, 275
 - misses, 514
 - organization, 218–19
 - parallel access to, 220
 - shutdown, 276
- transmission pipelines, 317
- average length of, 317
- transmission time, 315
- TRAP, 85
- traps, 84–6
- tree networks, 325–6
- schematics, 325
- true sharing misses, 271
- definition, 273
- TSO (total store order), 403
- turn model, 334–5
- two-level predictors
- classification, 121–2
 - concept of, 121
- UMA (uniform memory access), 450
- unconditional branches *see* jumps
- undefined instructions, 86
- unicast requests, 312
- uniform memory access (UMA), 450
- unloaded end-to-end packet latency, 315
- unloaded latency, 205
- update-based cache protocols
- bandwidth, 264
 - optimizations, 262–4
 - state-transition diagrams, 263
- UpgrAck, 280, 281, 282, 283
- upgrade complete (UpgrCompl), 283
- UpgrCompl (upgrade complete), 283
- user-level simulators, vs. full-system simulators, 490–1
- valid bit (V-bit), 216, 251, 252, 254, 453
- value communication, 375
- value predictions, 136–7
- in out-of-order processors, 137
- VAs *see* virtual addresses (VAs)
- V-bit (valid bit), 216, 251, 252, 254, 453
- VDD, 256
- vectorization, 163
- vector length (VL) registers, 159
- vector masks (VMs), 163
- vector microarchitectures, 158–64
- vector processors, 15–17
- characteristics, 75
 - definition, 158
- vector protocols, presence-flag, 278–9
- vectors
- addition, 14, 16, 105–6, 107–10, 159–60
 - basic block, 513–14
 - similarity between, 512–13
 - see also* presence-flag vectors (PFVs)
- vector strip mining, 161–3
- verification
- costs, 29
 - see also* design verification
- version management, 466
- vertical microcodes, 90
- very large scale integration (VSLI), 11
- design quality, 53
 - dynamic power in, 39
 - resistors, 38
- very long instruction word (VLIW)
- architectures, 141–3
 - advantages, 155
 - kernels, 143–4, 145, 146–51
 - size factors, 148
 - limitations, 155
 - parallelism in, 153
 - schematics, 142
 - use of term, 141
- very long instruction word (VLIW) microarchitectures, 140–57
- very long instruction word (VLIW) processors, hardware and, 75
- very long instruction word (VLIW) programs, loop unrolling, 143, 144
- very long instruction word (VLIW) scheduling, non-cyclic, 151–3
- vias
- definition, 29
 - electromigration, 65
 - see also* interconnects
- virtual-address caches
- with physical tags, 221–3
 - with virtual tags, 223–4
- virtual addresses (VAs), 212, 222, 274
- space layout, 213
- virtual address translation, 215
- virtual channels, 334–5
- virtual cut-through switching, 321–2
- virtual memory, 196, 212–24
- management, 194
 - motivations for, 212
 - operating system's view of, 213–15
 - organization, 213–14
 - system space, 194
- virtual memory hierarchy, 193
- virtual memory systems, 274
- virtual page numbers (VPNs), 215, 217, 219
- virtual pages, 274
- virtual space, of processes, 213
- virtual tags, virtual-address caches with, 223–4
- VL (vector length) registers, 159
- VLIW architectures *see* very long instruction word (VLIW) architectures
- VLIW (very long instruction word) microarchitectures, 140–57
- VLIW (very long instruction word) processors, hardware and, 75

542 **Index**

- VLIW (very long instruction word)
 scheduling, non-cyclic, 151–3
- VMs (vector masks), 163
- voltage, definition, 38
- voltage scaling, 47
see also dynamic voltage frequency
 scaling (DVFS)
- volts, 38
- VPNs (virtual page numbers), 215, 217,
 219
- VSLI *see* very large scale integration
 (VSLI)
- VTune, 515
- waiting, methods, 397
- WANs *see* wide area networks (WANs)
- WAR hazards *see* write after read
 (WAR) hazards
- warmup, concept of, 512–13
- Wattch, 499, 508
 schematics, 508
- watts, 38
- WAW hazards *see* write after write
 (WAW) hazards
- WB (write back), 92, 94, 96–7, 102–3
- weak ordering, 407–9
 orders in, 408
 speculative violations of, 414–15
- wearout, 64
 rapid, 37
- west-first routing algorithm, 335
- wide area networks (WANs), 13, 310
 routing algorithms, 330
- wide caches, 198–9
- WildFire, 287–8
- wire delays, 8
 impacts, 29
 linear, 29
 pipelining, 29
 technological issues, 28–9
- wires
 characteristics, **44**
 global, 29
 local, 29
 resistance, 28
- Wisconsin Wind Tunnel II, 507
- within-die variations, 70
 random, 70
 systematic, 70
- words, 87
- working sets, 196, 214
- workload behavior, projecting, 515–16
- workload characterization, 514–16
- workload sampling, 510–14
- work partitioning, 234
- wormhole switching, 322, 334
- write after read (WAR) hazards, 106,
 107–8, 111–12, 126, 475–7
 definition, 95
 on memory operands, 114
 solving, 141
- with rotating registers, 146–7
- write after write (WAW) hazards,
 107–8, 112, 126, 475–7
 definition, 95
 on memory operands, 114
 solving, 141
- write-allocate, 249
 policies, 271
- write-back caches, 204, 254
 architecture, 203
- write back (WB), 92, 94, 96–7,
 102–3
- write hits, 251, 262
- write misses, 251
- write policies, 203–4
- write register (WR), 94, 96–7
- write-run, definition, 263
- write-run length (WRL), 264
- write-run model, 263
- write-through caches, 203–4
 architecture, 203
- WRL (write-run length), 264
- WR (write register), 94, 96–7
- XactActive bit, 472–3
- XactMod bit, 472–3
- XactRead bit, 472–3
- XBusy, 473
- Xerox, 262
- XOR, 331
- Zesto, 491, 502