

Index

- accelerators, 12, 205, 230
 - attributes, 206
 - cryptography, 217, 219
 - data parallelism, 205, 208
 - deep neural networks (DNNs), 212
 - drawbacks, 217
 - efficient use, 208
 - Google tensor processing unit (TPU), 213
 - grain size, 208
 - graphics processing unit (GPU), 180, 205, 208, 219
 - machine learning, 212, 215, 219
 - media, 217, 219
 - narrow applicability, 208
 - new programming languages, 207
 - opportunities, 217
 - programming challenges, 207, 217, 219, 230
 - single-thread performance, 206, 230
 - why faster, 206
- Age of Accelerators, 230
- Age of New Computing Models, 231
 - higher-level architecture, 232
 - neuromorphic computing, 233
 - quantum computing, 233
- Age of Parallelism, continued, 229
- aircraft
 - Boeing 777, 64
 - Learjet 70/75, 64
- AMAT, 136
- AMD
 - graphics processing unit, 210
 - Rome multicore CPU, 181
- Apple iPhone 11 computer, 58, 120
- automobile
 - Ford Mustang, 64
 - Honda Civic, 64
 - Kia Forte, 64
 - Lamborghini Aventador, 64
- battery
 - electric vehicle, 64
 - lithium ion, 64
- Bengio, Yoshua, 220
- binary compatibility, 8, 16
- bit speed, hose and bucket, 53
- bits
 - definition, 50
 - natural size, 50
 - switching, 54
- book organization, 11
- Borkar, Shekhar, 102
- branch prediction, 91, 97, 101, 102, 162
 - two-bit, 97
- C++, 119
- cache, 123, 128, 150, 181
 - address filter, 123, 126–128, 132, 148
 - coherence, 181, 198
 - commercial design, 134, 164
 - correctness proof, 130
 - dynamic locality, 127, 131
 - hit, 129
 - hit rate, 129, 131
 - implementation, 132, 134, 164
 - inclusion, 198
 - Intel Skylake, 134, 164
 - least recently used, 127
 - lookup, 123, 126, 128, 132
 - map, 129
 - measuring locality, 142
 - miss, 129
 - miss rate, 129, 131
 - multi-level, 133, 134, 151, 164, 181, 198, 228
 - multicore, 180
 - performance, 131, 136, 145, 146
 - power consumption, 165
 - replacement, 127, 132
 - reuse, 126, 127
 - reuse distance, 142, 144
 - spatial locality, 127, 128, 152
 - tags, 129
 - temporal locality, 127, 128, 152
 - why work, 137
- capacitance, 54
- Cerebras, CS-2, 215
 - architecture, 215
 - machine learning accelerator, 215, 220
 - network, 215

- Cerebras, CS-2 (cont.)
 power, 215
 size, 215
charge pump, 54
Chien, Andrew A., 102
Clark, Jim, 220
clock period, 53, 58, 69
clock rate, 53, 69, 122
clock rate model, 55
clock speed, 58
clock speed model, 55
 example, 56
cloud computing, 176, 187, 228
 access latency pyramid, 190
 Alibaba, 187
 Amazon, 187
 application data parallelism, 186
 artificial intelligence (AI), 186
 Baidu, 187
 Clos network, 188
 cloud services, 187
 cooling, 188
 data analytics, 186
 datacenters, 188, 198
 efficient parallelism, 189, 197
 evolution, 194
 Facebook, 187
 function as a service (FaaS), 195, 198
 Google, 187
 grain size, 189, 197
 Hadoop, 186, 193, 198
 Hadoop file system (HDFS), 193
 higher-level abstraction, 195
 internet-scale services, 192
 large computations, 193
 machine learning, 186
 microservices, 194, 198
 Microsoft, 187
 networking, 188, 189, 198
 parallelism, 176, 187, 197, 198, 228, 229
 programming, 192
 racks, 188
 reliability, 194
 scale-out map-reduce, 193, 198
 server consolidation, 186
 serverless, 195, 198
 servers, 181, 188
 social media parallelism, 186, 198
 Spark, 186, 193, 198
 three-tier web services, 192, 198
 web search parallelism, 186
cloud server nodes, 120
Compaq portable computer, 58
Compaq Presario, 120
computability, 161
computer architecture, definition, 8
computer performance, explicit parallelism, 192
computer speed, 53
computer, basic elements, 17
 natural size, 50
 single chip, 59
 volume, 55
computing performance, 226
 accelerators, 12
 clock rate, 36
Computer History Museum, 172
database, 102
dynamic locality, 10, 12, 113, 126, 131, 152, 228
explicit parallelism, 10, 12, 176, 185, 198, 205,
 208, 228, 229
four pillars, 9, 226
growth, 6, 226
hidden parallelism, 9, 11, 36, 227
instruction-level parallelism (ILP), 36
miniaturization, 9, 11, 53, 57, 62, 102, 150,
 177, 227
sequential abstraction, 11
computing transforms, 1
 complexity classes, 170, 173
 cost, 6
 extraordinary characteristics, 6
 future of, 13, 226, 234
 general purpose, 12, 161, 168
 impact, 226
 performance, 6
 power, 6
 power consumption, 168
 size, 6, 62, 63
 universality, 170, 173
conferences
 ISCA, 235
 Rebooting Computing, 235
Cray, Seymour, 69
Cray-1 computer, 69, 120
critical path, 81
CUDA, 208, 220
 program example, 211
Dally, Willam, 220
DEC VAX750, 58
deep neural networks, size, 213
Dennard, Robert, 59, 69
Dennard scaling, 59, 60, 69, 227
 benefits, 62
 End of, 62, 176, 205
dependences, 81
 data, 80, 81
 flow, 81
 true, 81

- display, resolution, dots per inch (DPI), 63
DVD, bit size, 51
dynamic locality
 data access, 137
 instruction fetch, 137
 measurement, 142
 plots, 145
 portable, 145
 tuning, 146
dynamic random access memory (DRAM), 51
Eckert, J. Presper, 68
electric field, 61
ENIAC computer, 48, 57, 58, 68, 115, 120
Google, Tensor processing unit (TPU), 213
Google, TensorFlow language, 213
graphics processing unit
 AMD, 210
 architecture, 209
 CUDA, 208
 CUDA example, 211
 general purpose GPU, 220
 NVIDIA, 210
 OpenCL, 208
 parallelism, 209
 performance, 209, 210
 program example, 211
 programming, 208
 single instruction multiple thread (SIMT), 208
 SyCL, 208
 thread parallelism, 209
hidden parallelism, 80, 83
 renaming, 83
 superscalar, 102
higher-level architecture, 232
Hinton, Geoff, 220
hose and bucket, 55
HP Kayak XU, 120
IBM Mainframe 360, 58, 69, 115, 120
illusion, sequential abstraction, 80, 83
impact of computers
 automobiles, 3, 66
 commerce, 4, 66
 discovery, 5, 67
 home, 2, 66
 science, 5
 society, 2, 65, 226
input/output, 17
instruction execution
 compute, 34, 35
 decode instruction, 34, 35
 energy, 63
 fetch instruction, 34, 35
fetch operands, 34, 35
read memory, 34, 35
steps, 33
update program counter, 34, 35
write memory, 34
write registers, 34, 35
instruction pipelining, 36
instruction set
 architecture, 11, 16
 ARM, 17
 callee saves registers, 32
 caller saves registers, 32
 complex (CISC), 102
 IBM 360 Stretch, 102
 IBM POWER, 102
 Intel x86, 17, 29
 load–store, 113
 reduced (RISC), 29, 102, 113
 RISC-V, 19, 29
instruction-level parallelism (ILP), 36, 79, 81, 92, 97, 101, 162
 matrix multiply, 83
instructions
 computation, 18, 20
 control, 18, 20
 input/output, 18
 memory, 18, 22
integrated circuit (IC), 59
Intel Skylake memory hierarchy, 164
Intel Skylake processor, 161
Intel static random-access memory, 116, 151
Jacob, Bruce, 151
JavaScript, 119
 multicore parallelism, 186
 thread parallelism, 186
Kilby, Jack, 59
Kogge, Peter, 102
laptop, multicore, 180
Lecun, Yann, 220
logic gate, 60
 physical, 60
 power, 60
machine learning, 220
 accelerator, 212, 215, 219
 edge accelerators, 216
 low power, 216
machine learning accelerators, 220
MASPAR, 220
Mauchly, John, 68
McCalpin, John, 151
McKee, Sally, 151
memory, 17, 68
 average access time (AMAT), 136
 bandwidth, 122, 148, 150, 151

- memory (cont.)
 - cache performance, 145
 - capacity, 58, 68, 113, 120, 123, 150
 - capacity challenge, 121, 122
 - capacity growth, 119
 - consistency, 149
 - DRAM scaling, 116
 - dynamic locality, 12, 113, 126, 131
 - dynamic random access memory (DRAM), 116, 149, 151
 - future technologies, 118, 151
 - hierarchy, 113, 123, 133, 134, 136, 146, 151, 164, 181, 198, 228. *See also* cache
 - latency, 122, 123, 228
 - latency model, 121
 - magnetic core, 115, 151
 - measuring locality, 142
 - mercury delay lines, 114, 151
 - miniaturization, 113
 - non-volatile, 118, 151
 - ordering, 149
 - Ovonic, 118, 151
 - parallelism, 148
 - persistent, 118, 151
 - physical size model, 121
 - poor locality, 141
 - power consumption, 165
 - reducing latency, 123
 - reuse distance, 142, 144
 - semiconductor, 116, 151
 - sequential abstraction, 148
 - static random access memory (SRAM), 116, 151
 - technologies, 113
 - variable latency, 123
 - wall, 122, 151
- microprocessors, 59
- miniaturization, 49, 67
 - other technologies, 64
- models
 - memory physical size and latency, 121
 - physical size and clock period, 56, 70
- Moore, Gordon, 59, 69
- Moore's Law, 59, 69, 227, 229
- multicore, 180, 197, 198, 228
 - AMD Rome, 181
 - commercial example, 162, 181
- multiple-issue, 79, 101
- network, 17
- neuromorphic computing, 233
- Nickolls, John, 220
- Noyce, Robert, 59
- number of computers, 2
- Nvidia, 210
 - Fermi architecture, 209
- Old North Church, 51
- OpenCL, 208
- OpenMP, 183
 - matrix multiply, 185
- out-of-order, 79, 88, 163
- Papal smoke, 51
- parallel computing, 177
 - parallelism, 187
 - supercomputers, 187, 197
- parallelism
 - hidden, 76
 - instruction level (ILP), 76
- pipelining, 76, 79, 101, 102, 163
- Pollack, Fred, 198
- portability, software, 8, 16
- power
 - consumption, 63
 - scaling, 63, 177
 - scaling challenge, 59, 60, 177
- power wall, 59, 60
- processor, 17
 - Apple A13 Bionic, 63
 - ARM Thumb, 49
 - ARM Thumb power, 49
 - ARM Thumb size, 49
 - clock rate, 122
 - commercial, 161
 - commercial performance, 172
 - core, 161
 - Cortex-M3, 49, 57
 - ENIAC, 63
 - general purpose, 172
 - Intel Skylake, 161, 180
 - Intel x86, 161
 - multicore, 162, 176, 180, 197
 - parallelism, 228
 - Passmark benchmark, 169
 - power consumption, 168, 172, 177, 230
 - single-thread performance, 169, 177
- processor information
 - Computer History Museum, 172
 - Hot Chips Symposium, 172
 - Microprocessor Report, 172
- programs
 - accelerators, 217
 - array of structs, 26, 29
 - arrays, 26
 - basic expressions, 23
 - bubble sort, 76
 - cache performance, 145
 - cloud computing, 192. *see also* cloud computing, programming
 - conditionals, 27
 - data structures, 24
 - design for locality, 137

- dot product, 81
dynamic locality, 126, 131, 150, 152, 228
explicit parallelism, 185
iteration, 27
linear speed-up, 177, 198
loop, 29
matrix multiply, 83, 92
measuring locality, 142
multi-dimensional arrays, 26
objects, 24
parallel, 177, 183, 197
parallel programming, 185, 197
parallel speed-up, 177, 198
parallelism, 197
poor locality, 141
procedure calling convention, 31
procedure calls, 30
reuse distance, 142, 144, 151
spatial locality, 137, 228
stack, 30
struct of arrays, 26
sublinear speedup, 177
temporal locality, 137, 228
pthreads, 183
Python, 119
 multicore parallelism, 186
 thread parallelism, 186
quantum computing, 233
R
 multicore parallelism, 186
 thread parallelism, 186
registers, 19
renaming, 83, 101, 162
 algorithm, 87
 registers, 85
 reorder buffer (ROB), 88, 93, 163
 variables, 85
reuse distance, 142, 152
 bubble sort, 143
 dynamic locality, 144
 examples, 144
 plots, 145
 portable, 145, 151, 152
 tuning, 146
Rust, 119
scientific principles, 1, 11
sequential abstraction, 35, 76, 83, 85, 101, 176
 program composition, 78
 security, 99
server, commercial example, 181
server, multicore, 180
Shannon, Claude, 50, 68
software
 accelerators, 217
 dynamic locality, 152
 memory growth, 119, 121, 228
 memory requirements, 119, 121, 150
 mobile devices, 119
 parallel, 177
 portability, 230, 231
solid-state disks, 53
space-time warping, 123
speculative execution, 79, 90, 93, 97, 101, 162
 Meltdown, 99
 security risks, 99
 Spectre, 99
speed of light, 69
speed-up, 177
storage, 17
supercomputing, 187, 197, 228
 father of, 69
superscalar, 162
swift, 119
SyCL, 208, 220
Thinking Machines, 220
thread parallelism, 197, 198
 efficient parallelism, 189, 197, 198
 grain size, 189, 197, 198
 loops, 184
 matrix multiply, 185
 OpenMP, 183
 OpenMP example, 183
 pthreads, 183
 pthreads example, 184
 server, 185, 197
 thread safety, 184, 185
tiling, 146
Tomasulo, Robert, 102
TPU, 220
 architecture, 213
 network, 213
transmission line, 54
trip counts, 92
TTL (transistor–transistor logic), 58
Unity, 119
UNIVAC computer, 48, 53, 58, 115, 120
 power, 48
 size, 48
 weight, 48
University of Pennsylvania, 48
Unreal Engine, 119
vacuum tubes, 58
von Neumann bottleneck, 113, 122
wind turbines, 64
Wulf, William, 151