

# Index

## A

- Algebraic transformations, 18–19
- Alias, 132, 187
- Automaton, 20, 89–93, 98, 100, 103–107, 110, 129–132, 142, 143, 238

## B

- Balance equations, 56–58, 62, 63, 67, 73
- Basic blocks, 127, 128
- Bit vector, 137, 138
- Blocking, 22, 25, 26, 28, 34, 48, 49, 80, 81, 87–101, 103–105, 115, 123, 124, 126–130, 132, 137–139, 187, 230, 231, 234, 236, 238, 244, 245
- Buffer sharing, 143, 144

## C

- C, vi, 3, 14, 21, 25–28, 30, 31, 33–34, 48, 96, 99, 121–124, 126, 127, 130–132, 136, 138–141, 193, 239
- C++, 3, 21, 25, 31, 123, 147
- Call tree, 137, 139
- Causality, 47, 109, 159, 177, 203, 210, 231, 233–235, 238
- Cell processor, 140, 145
- Channels, 36, 42–45, 48, 50, 122–130, 132, 133, 136–144, 176, 178, 179, 197, 253
- Clock analysis, 230, 231, 236, 244
- Clock calculus, 8, 18, 25, 28, 212, 214, 223, 259, 260
- Clocks, viii, ix, 2–9, 12, 14–21, 23, 25–35, 46–49, 56, 95, 100, 101, 103, 108, 110, 114, 122, 123, 149–152, 159, 163–166, 173–176, 201–226, 230–232, 234, 236–238, 241, 244–251, 259, 260

- Code generation, vi, viii, 3, 8, 10, 13–39, 81, 85, 86, 93, 94, 117, 123–132, 137, 138, 143, 160, 181, 183, 188, 193, 194, 203, 229, 230, 236
- Communication, vii, 2, 5, 8, 17, 18, 22–24, 26, 27, 30, 35–37, 39, 41–43, 45–50, 54, 58, 59, 63, 69, 73, 74, 80, 83, 84, 91, 110–114, 117, 122, 123, 126, 129, 132, 133, 135–139, 141, 143, 145, 166, 179, 182, 203, 205, 215, 217–219, 231, 232, 236, 238, 239, 241, 244, 246, 248–250, 253, 255–260
- Communication patterns, 129
- Compilation, vi, vii, 1–39, 41, 43, 51, 58, 59, 64, 73, 74, 94, 100, 114, 115, 148, 173, 202, 226, 230, 232, 235, 237
- Components, v, viii, 1, 8, 25, 26, 30, 35, 36, 38, 42, 43, 50, 51, 56, 60, 80, 83, 88, 89, 94, 96–97, 114–116, 147, 148, 151, 152, 159, 160, 165, 167, 168, 170, 182, 201, 204, 205, 215–217, 233, 234, 239
- Compositional approach, 142
- Concurrency, 30, 41, 42, 47, 56, 58, 60, 69, 73, 74, 81, 124, 141, 197, 236
- Condition variables, 121
- Contract
  - assume/guarantee, 151–161, 167
  - boolean algebra, viii, 148, 153, 155, 156, 158, 160, 169
  - design, viii, 147, 148, 158, 168–170
  - process algebra, 148, 152–163
  - real-time, 147–149
  - refinement, 147, 156–159, 161–162, 167, 168
  - semantic, 152, 154, 161, 162

specification, 112, 114, 117, 147–149, 151, 152, 159, 160, 163, 168–170  
 synchronous language, 148, 160  
 Contract-directed code generation, 160  
 Control-flow graph, 127, 202  
 Control software, 80, 110, 113, 117  
 Correct by construction static routing transformations, 60  
 CSP, 44, 122  
 Cyclo-Static DataFlow (CSDF), 44, 45, 59–60, 62, 73

**D**

Data Control Graph (DCG), 3, 14, 20, 21, 34  
 Dataflow, 41–44, 46–60, 74, 81, 86, 88–93, 96, 98–101, 105, 107, 109, 122, 230, 231, 236–238, 241–246, 248, 249, 251–253, 258–260  
 Dataflow graphs, 59–60, 242, 251  
 Deadlock, 4, 9, 14, 15, 18, 19, 25, 43, 50, 51, 57–60, 63, 68, 71, 73, 126, 130, 141–143, 182–184, 197, 225  
 Determinism, viii, 74, 121–123, 136, 248  
 DFS, 127  
 DMA, 145

**E**

Embedded software, vi–viii, 83, 173–198  
 Embedded software design, vi  
 Embedded systems, v, vii, viii, ix, 1–4, 14, 37, 41–75, 111, 121, 147, 148, 160, 175, 176, 201–205, 226, 229, 230, 235, 241  
 Endochrony, 14, 183, 190, 197  
 Endochrony (endochronous system), 14, 24, 183, 190–192, 197, 260  
 Esterel, 21, 47, 122, 135, 173–176, 236, 237  
 Event-driven, 175  
 Event function, 17, 138, 139  
 Exception handler, 137  
 Exceptions, 132–139, 144, 256  
 Exceptions and communication, 135–138, 144  
 Exception scope, 136  
 Explicit model-checking, 142  
 Exponential state space, 143  
 External variable, 86, 88, 94, 99, 111, 113–144

**F**

FIFO, 4, 24, 49, 50, 66, 134, 135, 145, 166, 217  
 Fixed point, 129, 173

Flight software, 79–85, 87, 89, 92, 93, 98, 99  
 Formal methods, vi–viii, 116  
 Formal verification, vi, 2, 37, 48, 49, 74, 75, 81, 122, 148, 230, 235, 237  
 Function calls, 23, 31, 32, 81, 126, 132, 133, 140  
 Function pointers, 124, 126, 132

**G**

Globally asynchronous and locally synchronous (GALS), 2, 20, 80, 84, 88, 113, 116

**H**

Hardware/software boundary, 122  
 Haskell, 145

**I**

IBM, 140, 145  
 Inferred parameters, 123  
 Interconnect optimization, 42–43, 75  
 Interleavings, 122  
 I/O library, 144

**J**

Java, 3, 25, 31, 38, 96, 121, 147

**K**

Kahn networks, 122, 208, 237  
 K-periodic routing, 45, 59–74  
 K-periodic throughput equalization, 51, 59, 63–64

**L**

Latency insensitive design (LID), 43, 46, 48–49, 56, 58, 74  
 Locks, 68, 94, 121, 136, 138, 211  
 Logical clock, viii, ix, 2, 47, 49, 149, 201–226  
 Lustre, 48, 82, 122, 159, 173, 176, 214, 236, 237, 241–243, 259

**M**

Macros, 127  
 Marked Graphs, 43, 44, 46, 49–56, 58, 64, 68, 74  
 Mask operation, 137

Model-checking, 48, 142, 143, 162, 235  
 Model-Driven Engineering (MDE), 80, 82, 85, 87, 116, 117, 201  
 Models of computation, vi, 42, 43, 74, 160, 173  
 Modulo scheduling, 202  
 Multi-clock, 37, 174, 236, 260  
 Multiway rendezvous, 133, 136, 138, 144

## N

Nested loops, 127, 202  
 Network block, 123  
 Next, 125–128, 132, 133, 136, 142  
 Non-determinism, viii, 121, 122  
 NuSMV, 141, 142

## O

Offline scheduling, 231  
 Operating system, 24–27, 80, 121, 124, 139, 232, 239  
 Optimality condition, 255  
 Optimization, 2, 37, 38, 43, 45, 48, 50, 55, 56, 58, 122, 176, 230–232, 234, 241, 244, 247–249, 254–260

## P

Parallel exceptions, 135–136  
 Parallelism, 41, 74, 121, 239  
 Par construct, 132  
 Pass-by-reference, 132, 140  
 Pipeline, 55, 143, 201, 220  
 Point-to-point channels, 42, 124  
 Poisoned, 136–141, 144  
 Polychrony, vi, viii, 3, 9, 14, 17, 18, 21, 25–27, 30, 31, 34, 37–39, 86, 117, 176, 178, 190, 236, 259, 260  
 Polychrony, multi-rate, 173  
 Porting C programs, 132  
 POSIX threads, 136, 139  
 PPU, 140  
 Prime implicate, 176, 189–192, 197, 198  
 Processes, 122–127, 129–135  
 Process Networks, vii, 41–45, 74, 201, 202, 222  
 Product machine, 129  
 Program analysis, vi–viii, 3, 143  
 Program counter, 124, 129, 130  
 Program transformation, 3, 37, 38  
 Pthreads, 121, 140

## R

Races, viii, 121  
 Reader pointer, 124  
 Real-time, v–vii, ix, 3, 8, 43, 75, 80–82, 85, 95, 99, 111, 112, 147–149, 198, 201–206, 226, 229–260  
 Real-time scheduling, 229–232, 237, 239–241, 252, 258, 259  
 Reconfiguration, 86, 111, 112, 114–117  
 Recursion, 123, 124, 132–137  
 Recv, 132–134, 136, 137, 141, 142, 144  
 Refinement, vi, 2–4, 12, 19–21, 29, 86, 88, 96–99, 118, 147, 156–159, 161, 162, 167–169  
 Rendezvous, 122, 123, 126, 133, 135, 136, 138–139, 143–145  
 Resuming, 102, 103, 105, 124, 126, 132

## S

Scheduler, 9, 25, 26, 30, 31, 38, 88, 112, 117, 124, 126, 139, 205, 240  
 Scheduling, vii, viii, ix, 20, 23–24, 26, 29–31, 33, 38, 39, 41, 43, 45, 48, 49, 57, 64, 74, 82, 127, 129, 132, 141, 143–145, 190, 193, 197, 202, 209, 222, 229–233, 235, 237, 239–241, 248, 252–260  
 Scheduling-independent, 122, 135, 144, 145  
 Scheduling policy, 121, 130, 240  
 Semantics, vi, 1–4, 7, 9–10, 21, 22, 25, 42, 44, 45, 69, 81, 86, 88, 99–111, 117, 122, 123, 135, 136, 144, 152, 161, 162, 176, 177, 182, 203, 204, 206, 208–210, 216, 217, 221–226, 230, 232, 234, 235, 243–244  
 Send, 125, 126, 130, 132–138, 141, 142, 144  
 Sequential implementability, 183–185, 190–193, 197  
 Service oriented middleware, 110–111  
 Shared memory, viii, 121, 143  
 Shared variables, 8, 12, 22, 86, 94, 122, 233  
 SHIM, vii, 121–145  
 SIGNAL, vi, vii, viii, 2–32, 34–38, 41, 42, 47, 48, 56, 75, 80–82, 86, 88, 94, 95, 98–109, 114, 122, 123, 147–170, 173–194, 197, 205, 214, 230, 233–239, 248, 250–260  
 Software architecture, vi, 80, 82, 87–96, 116  
 Software synthesis, viii, ix, 173–198  
 Spanning tree, 127  
 SPIN, 122, 141  
 State abstraction, 129, 143  
 State names, 130

State signature, 129  
 Static routing, 60  
 Static scheduling, ix, 26, 29–30, 38, 41, 127,  
 143, 222, 231, 248, 253, 257  
 Subset construction algorithm, 129  
 Switch statement, 132  
 Synchronous Data Flow (SDF), vi, viii, 43–46,  
 56–60, 62, 63, 67, 73, 74, 86, 197,  
 203, 204, 221–225, 231, 241  
 Synchronous Elastic Systems, 43, 48  
 Synchronous formalism, 230, 231, 234, 235,  
 237, 242  
 Synchronous islands, 88, 93–94, 99, 112, 113,  
 116, 117  
 Synchronous languages, vi, viii, 2, 88, 148,  
 160, 176, 202, 230, 232, 235–237  
 Synchronous model, ix, vi, 56, 114, 122, 141,  
 231  
 Synchronous programming, vi, 173, 197, 237  
 SynDEx, 3, 48, 237, 241–249, 251–253, 259,  
 260  
 Synergistic processing unit (SPU), 140  
 Synoptic, vii, 79–118  
 Synthesized code, 124, 125

**T**

Tail-recursion, 124  
 Termination, 125, 126, 130  
 Testing, 84, 121  
 Transitive poisoning, 136  
 Turing complete, 122

**V**

Verification, vi, vii, 2, 3, 37, 48, 74, 75, 81, 82,  
 85, 86, 88, 89, 117, 121, 122, 148,  
 182, 189, 194, 232, 235, 237, 248,  
 251  
 Verilog, 48, 123

**W**

Writer pointer, 124

**X**

X10, 145