

Circulation of this  
edition outside the  
Indian subcontinent is  
**UNAUTHORIZED**

# Computer Systems Design and Architecture

Second Edition

**NEW!**

Chapters on  
System Software  
Architectures and Parallel  
Processing. Coverage of  
ARM, Blackfin and  
Pentium Architectures.

Vincent P. Heuring  
Harry F. Jordan  
T. G. Venkatesh

ALWAYS LEARNING

PEARSON



## Trademark Acknowledgments

8008, 8080, 8086, Intel, Pentium, and P6 are registered trademarks of the Intel Corporation.

68000, 68010, 68020, 68040, and PPC601 are trademarks of the Motorola Corporation.

Alpha, DECnet, PDP-11, VAX, and VAX11/780 are trademarks of the Digital Equipment Corporation.

AppleTalk, LocalTalk, Macintosh, Macintosh Quadra, FireWire and QuickDraw are registered trademarks of Apple Computer, Inc.

Centronics is a registered trademark of the Centronics Data Computer Corporation.

Espresso is a registered trademark of Berkeley Systems, Inc.

FAST is a registered trademark of National Semiconductor in France, Italy, Canada, the United States, and the Benelux countries.

HPGL and PCL are registered trademarks of the Hewlett-Packard Company.

IBM, 360, and System/360 are trademarks of the International Business Machines Corporation.

Maxtor and LARAMIE are trademarks of the Maxtor Corporation.

MIPS, R3000, and R4000 MIPS are trademarks of MIPS Technology, Inc.

NetWare is a registered trademark of Novell, Inc.

NuBus is a trademark of Texas Instruments, Inc.

PostScript is a registered trademark of Adobe Systems, Inc.

SPARC and UltraSparc are registered trademarks of SPARC, International Inc., licensed to Sun Microsystems, Inc.

Tri-state is a registered trademark of National Semiconductor in member nations of the Madrid Convention.

The UNIX trademark is licensed exclusively through the X/Open Company Ltd.

VITESSE is a registered trademark of the Vitesse Semiconductor Corporation.

Windows is a registered trademark of the Microsoft Corporation.

XNS is a trademark of the Xerox Corporation.

## Contents

	<b>Preface</b>	<b>ix</b>
<b>CHAPTER 1</b>	<b>The General Purpose Machine</b>	<b>1</b>
	1.1 The General Purpose Machine	2
	1.2 The User's View	3
	1.3 The Machine/Assembly Language Programmer's View	5
	1.4 The Computer Architect's View	11
	1.5 The Computer System Logic Designer's View	16
	1.6 Historical Perspective	19
	1.7 Trends and Research	23
	1.8 Approach of the Text	24
	Summary	25
	Bibliography	26
	Exercises	26
<b>CHAPTER 2</b>	<b>Machines, Machine Languages, and Digital Logic</b>	<b>29</b>
	2.1 Classification of Computers and Their Instructions	30
	2.2 Computer Instruction Sets	32
	2.3 Informal Description of the Simple RISC Computer, SRC	47
	2.4 Formal Description of SRC Using Register Transfer Notation, RTN	55
	2.5 Describing Addressing Modes with RTN	64
	2.6 Register Transfers and Logic Circuits: From Behavior to Hardware	66
	Summary	77
	Bibliography	78
	Exercises	79
<b>CHAPTER 3</b>	<b>Some Real Machines</b>	<b>83</b>
	3.1 Machine Characteristics and Performance	84
	3.2 RISC versus CISC	88

	3.3	A CISC Microprocessor: The Motorola MC68000	93
	3.4	A RISC Architecture: The SPARC	117
	3.5	More CISC and RISC Processors	132
	3.6	Digital Signal Processors	137
		Summary	141
		Bibliography	142
		Exercises	143
<b>CHAPTER 4</b>		<b>Processor Design</b>	<b>147</b>
	4.1	The Design Process	148
	4.2	A 1-Bus Microarchitecture for the SRC	149
	4.3	Data Path Implementation	154
	4.4	Logic Design for the 1-Bus SRC	155
	4.5	The Control Unit	167
	4.6	The 2- and 3-Bus Processor Designs	175
	4.7	The Machine Reset	181
	4.8	Machine Exceptions	183
	4.9	Microprogramming	191
		Summary	205
		Bibliography	206
		Exercises	206
<b>CHAPTER 5</b>		<b>Processor Design—Exploiting Parallelism</b>	<b>211</b>
	5.1	Pipelining Overview	212
	5.2	Design of Linear Pipeline	214
	5.3	Design of Static and Dynamic Multifunction Non-Linear Pipeline	221
	5.4	Design of Instruction Pipeline	239
	5.5	Pipeline Hazards	256
	5.6	Instruction-Level Parallelism	280
		Summary	303
		Bibliography	303
		Exercises	304
<b>CHAPTER 6</b>		<b>Computer Arithmetic and the Arithmetic Unit</b>	<b>311</b>
	6.1	Number Systems and Radix Conversion	311
	6.2	Fixed-Point Arithmetic	322
	6.3	Seminumeric Aspects of ALU Design	356
	6.4	Floating-Point Arithmetic	362
		Summary	371
		Bibliography	371
		Exercises	372
<b>CHAPTER 7</b>		<b>Memory System Design</b>	<b>377</b>
	7.1	Introduction: The Components of the Memory System	378
	7.2	RAM Structure: The Logic Designer's Perspective	381

	7.3	Memory Boards and Modules	397
	7.4	Memory Hierarchy	416
	7.5	The Cache	422
	7.6	Virtual Memory	444
	7.7	The Memory Subsystem in the Computer	465
		Summary	467
		Bibliography	468
		Exercises	468
<b>CHAPTER 8</b>		<b>Input and Output</b>	<b>473</b>
	8.1	The I/O Subsystem	474
	8.2	Programmed I/O	477
	8.3	I/O Interrupts	486
	8.4	Direct Memory Access (DMA)	494
	8.5	I/O Data Format Change and Error Control	497
		Summary	503
		Bibliography	504
		Exercises	504
<b>CHAPTER 9</b>		<b>System Software Architectures</b>	<b>507</b>
	9.1	Operating System	507
	9.2	Compilers, Loaders, and Linkers	521
	9.3	Assembly and Assemblers	525
		Summary	535
		Bibliography	535
<b>CHAPTER 10</b>		<b>Peripheral Devices</b>	<b>537</b>
	10.1	Magnetic Disk Drives	538
	10.2	Improving Disk System Performance and Reliability	547
	10.3	Other Mass Storage Devices	549
	10.4	Display Devices	550
	10.5	Printers	558
	10.6	Input Devices	560
	10.7	Interfacing to the Analog World	561
		Summary	565
		Bibliography	566
		Exercises	566
<b>CHAPTER 11</b>		<b>Communications, Networking, and the Internet</b>	<b>569</b>
	11.1	Computer to Computer Data Communications	570
	11.2	Serial Data Communications Protocols	578
	11.3	Local Area Networks	584
	11.4	Modern Serial Buses: USB and FireWire	587
	11.5	The Internet	591

	Summary 601	
	Bibliography 602	
	Exercises 603	
<b>CHAPTER 12</b>	<b>Parallel Processing</b>	<b>605</b>
	12.1 Taxonomy of Parallel Architectures 606	
	12.2 Vector Supercomputers 607	
	12.3 SIMD Array Processors 609	
	12.4 Shared Memory Multiprocessor 611	
	12.5 Interconnection Networks (IN) 612	
	12.6 The Systolic Array Processors 619	
	Bibliography 625	
	Exercises 625	
<b>APPENDIX A</b>	<b>Digital Logic</b>	<b>627</b>
	A.1 Combinational Logic 628	
	A.2 Truth Tables 628	
	A.3 Logic Gates 630	
	A.4 Properties of Boolean Algebra 634	
	A.5 The Sum-of-Products Form and Logic Diagrams 636	
	A.6 The Product-of-Sums Form 638	
	A.7 Positive versus Negative Logic 640	
	A.8 The Data Sheet 641	
	A.9 Digital Components 643	
	A.10 Reduction of Two-Level Expressions 652	
	A.11 Speed and Performance 658	
	A.12 Sequential Logic 662	
	A.13 J-K and T Flip-Flops 667	
	A.14 Design of Finite State Machines 669	
	A.15 Mealy versus Moore Machines 676	
	A.16 Registers 677	
	A.17 Counters 680	
	Bibliography 680	
	Exercises 681	
<b>APPENDIX B</b>	<b>RTN Description of SRC</b>	<b>687</b>
	B.1 SRC Without Reset or Exceptions 687	
	B.2 Additions to SRC for Reset and Interrupts 689	
	B.3 Unified RTN for SRC 690	
	B.4 Register Transfer Notation—RTN 693	
	B.5 SRC Assembly Language Conventions 694	
<b>APPENDIX C</b>	<b>Selected Problems and Solutions</b>	<b>697</b>
	<b>Index</b>	<b>713</b>

---

### To the Instructor

---

This book is suitable for an introductory course on computer design at the junior, senior, or introductory graduate level. We assume that the student has had at least an introductory course in some high-level programming language such as C or Pascal, and a semester of logic design. However, a comprehensive appendix on digital logic design, written by Professor Miles Murdocca of the Internet Institute USA, provides sufficient background material for teaching the course to students without previous digital design experience.

appendix on  
digital logic  
design

Appropriate topics for such a book have changed considerably in recent years, as desktop computers have evolved from simple, stand-alone units into complex systems attached to high-speed networks and internetworks. Earlier generations of microprocessors had almost trivial internal structure. Present designs contain multiple pipelined functional units with support for multiple processors and memories. Areas of computer design and architecture that were barely touched upon in the not-so-distant past have become major topics for discussion. Introductory compiler courses now routinely discuss optimization for pipelined processors. Users worry about whether they should add level-2 cache memory to their PCs. Support personnel wearily try to explain to the computer user how to configure the subnet mask for their network slip connection.

The topics of pipelined processor design, the memory hierarchy, and networks and internetworking are moving to center stage in the arena of computer design and architecture. Therefore we devoted the major parts of three chapters to treatment of these subjects.

pipelined  
processors  
memory  
hierarchy

Given the focus on computer design and computer architecture, we approach the study of the computer from three viewpoints: the view of the assembly/machine language programmer, the view of the logic designer, and the view of the system architect.

networking and  
the Internet  
three views  
of the general  
purpose  
machine

In covering the topic of gate-level computer design, we follow a model architecture through the design process, from the instruction set design level to the processor design level. Given the choice of using either a commercial machine with all of the complicating features that are necessary to make such a machine commercially successful, or using

**Numerics**

- 1's complement 317
- 1-address (accumulator) machines 41
- 1-1/2 address machines 44
- 2's complement 317
- 160, 158
- 2-address machines 40
- 3-address machines 40
- 4-address machines 38

**A**

- absolute addressing 101
- abstract RTN 76, 148
- accumulator register 30
- accumulator, in arithmetic operations 340
- accumulator-based machines 30, 41
- adders
  - carry lookahead 327
  - ripple-carry 326, 651
- addition
  - fixed point 322
  - unsigned 323
- addition and subtraction
  - floating point numbers 366
  - integer hardware 324
- address as data type 36
- address translation
  - in virtual memory 448
- addressing modes
  - absolute 65
  - as access paths to operands 45
  - autoincrement and autodecrement 66
  - based 47, 66
    - direct 45, 65
    - displacement 47, 65
    - immediate 45, 65
    - indexed 47, 65
    - indirect 46, 65
    - register 65
    - register indirect 47, 65
    - relative 47, 65
    - RTN description of 64
- ALU
  - branching hardware 356–359
  - condition codes 357
  - hardware design of 362
  - shift and rotate hardware 359
- ALU instructions
  - examples of 36
  - operand access in 36
- analog interfaces 561–565
  - analog to digital converters (ADCs) 562–575
  - digital to analog converters (DACs) 561–562
  - errors in 564
    - gain error 565
    - missing codes 565
    - monotonicity 565
    - offset error 565
- analytical engine 20
- Apple Computer 23
  - Cray computer used to emulate 15
- Apple PowerMacintosh G4 13
- application layer, in communications 577
- ARPA 569
- ARPANET 569
- ASCII code 10, 582
  - table of 583

assembler  
 capabilities of 529  
 location counter 533  
 pseudo operations 529  
   DS, DC, EQU, ORG 530  
 symbol table 532  
 assemblers, two pass 534  
 assembly 525  
   process of 528–530  
 assembly language programming  
   why? 526  
 associative mapped caches 423  
 asynchronous data transmission 580  
 asynchronous I/O 480  
 autoincrement and autodecrement addressing 100

**B**

Babbage, Charles 20  
 Babel  
   tower of 33  
 bandwidth, burst 475  
 bandwidth, memory 381  
 Bardeen, John 22  
 barrel shifter 360  
 base conversions 316  
 based addressing 47, 100  
 baud vs. bps 582  
 baudot code 582  
 Baudot, J-M-E 581  
 Bell Laboratories 20  
   Model I Computer 20  
 big endian storage 380  
 bit error rate  
   defined 498  
   in communications 575  
 bit ORing 203  
 Boole, George 20  
 Boolean algebra 20  
 Booth recoding 349  
 branch targets and target addresses 37  
 branches  
   conditional 37  
   delayed 91, 274  
 branching conditions  
   ALU hardware for 356–359  
   calculation of, in MIPS R2000 358  
 Brattain, Walter 22  
 bridges, LAN 574  
 burst bandwidth 475  
 bus  
   as network topology 573  
   concept of 71  
   propagation delay 169

## buses 13

examples of (Figure) 13  
 FireWire 587–591  
 I/O, structures of 475  
 multiple, in processor design 175  
 parallel 12  
 PCI 14  
 in pipelined designs 240  
 rules of 75  
 serial 13  
 tri-state 72  
 USB 14, 587–591  
 wired OR 72  
 busy-wait loop 117  
 byte addresses 34  
 byte ordering 379  
 byte, defined 2

**C**

cache memory 15, 422–444  
 associative mapping 423  
 block set associative 426  
 cache line 422  
 direct mapping 425–426  
 dirty bit 433  
 hardware requirements for 429  
 in two level hierarchy 444  
 Intel Pentium cache 429  
 mapping function 422  
 memory address fields 422  
 performance 440  
 PPC G4 example 441  
 primary and secondary levels in 421  
 random block replacement 431  
 read miss policies 439  
 set associative caches  
   See block set associative caches 426  
 valid bit 423  
 write allocate 433  
 write back policy 432  
 write miss policy 440  
 write policies 435  
 write through 432  
 carry lookahead adders 328  
 CD-audio data storage 549  
 CD-ROM drives 549  
 Centronics interface 486  
 channel, DMA 496  
 character generators 554  
 character I/O 481

CIDR 598  
 circuit switching, networks 573  
 CISC 9  
   design philosophy 90  
   MC68000 as 93  
 classification of machines  
   load-store 44  
   memory-memory 44  
   register-memory 44  
   register-to-register 44  
   see also  
     general register machines  
     stack machines  
 clocking  
   clock skew 169  
   estimating minimum clock period 168  
   rise and fall times 169  
 clocking and timing 167  
 CMOS ROM 395  
 Cocke, John 8  
 communications, computer 570–601  
   See also  
     Ethernet;  
     Internet; LANs;  
     networks;  
     RS-232 serial comm.  
   addressing and routing in 576  
   application layer 577  
   ASCII code in 582  
   baseband vs. broadband 572  
   baud vs. bps 582  
   bit error rate 575  
   channels 570  
   data link layer 577  
   data rates in 581  
   error detection and correction in 576  
   flow control in 576  
   frequency-division multiplexing 571  
   layer models 577  
     OSI 577  
   levels and layers in 575  
   media for 575  
   modems 581–582  
   network layer 577  
   OSI layer model 577  
   physical layer 577  
   presentation layer 577  
   protocols 570  
     RS-232 578–581  
   RS-232 serial 578–581  
   session layer 577  
   signal encoding methods 572  
     Manchester encoding 572  
     NRZ encoding 572  
     NRZI encoding 572

synchronization in 576  
 tasks in 575  
 time-division multiplexing 571  
 transport layer 577  
 compact disk 549  
 comparator 681  
 compiler  
   pipeline hazard detection by 263  
   writers 11  
 compilers 10  
   role in RISC machines 92  
 complement number systems 317–321  
   addition and subtraction hardware 356  
 complement, 1's 317  
 complement, 2's 317  
 computer  
   data path 76  
   front panel 21  
 computers  
   analytical engine 19  
   Bell Laboratories Model I 20  
   generations of 21  
   relay computer 20  
 concrete RTN 76, 148  
 condition codes  
   calculation of in ALU 357  
 conditional branches 48  
 control sequences in processor design 162  
 control unit  
   clocking logic 173  
   control signal encoder 171  
   control step generator 172  
   hardwired 171  
   implementation 175  
   role in system initialization 174  
 cookie crumbs  
   Hansel and Gretel 112  
 core  
   dump 21  
   memory 22  
     usage of term as sign of age 22  
 cost  
   cost/performance tradeoffs 84  
   ICs 84  
 CPU-memory interface 378  
 CRC (cyclic redundancy check) 502  
 CRT, cathode ray tube 551  
 cylinder, disk drive 541  
**D**  
 DARPA 569  
 data link layer, in communications 577  
 data movement instructions 35  
   examples 35

- data path 76
  - defined 148
  - timing in 167
- data transmission
  - asynchronous 588
  - isochronous 588
- data types
  - addresses as 36
- data typing
  - at the machine level 10
  - type checking
    - at the machine level 10
- DC, assembler pseudo operation 529
- DDR SDRAM 392
  - Micron Technology 521 Mb 393
- debugger 4
- decoders, tree and matrix 386
- delayed branches
- delayed loads and stores 91
- demand paging 447
- dependence
  - between instructions in pipeline designs 240
- development system 19
- device drivers, input 483
- DHCP 598
- Dhrystones (performance measurement) 88
- Digital Equipment Corporation, DEC
  - Alpha 21164, superscalar processing 285
  - VAX11 7
- digital to analog converters (DACs) 561–562
- diminished radix complement 316
- direct addressing 45, 66
- direct mapped caches 425–426
- direct memory access
  - see DMA 494
- disassembly 525
- disk cache 466
- disk drives 538–547
  - access times 545
  - capacity calculations 544
  - cylinders 544
  - data organization in 541
  - dynamic properties 544
  - formatting 543
  - IBM 75GXP 547
  - operating system interface to 543
  - platters 539
  - RAID arrays 547–548
  - read-write heads 539
  - SMART 548
  - tracks and sectors 541
  - zone-bit recording 546
- dispersion, signal 169
- displacement addressing 47, 65
- Display Codes, CDC 10
- display devices 550–557
  - flat panel displays 557
  - memory mapped video 553–554, 556
  - video display terminals 553–557
- division
  - floating point 367
  - integer 352–355
- DMA 494–497
  - channels and I/O processors 497
  - hardware for 496
- domain names, in the Internet 591
- dot-matrix printers 559
- DRAM 382, 390–392
- DS, assembler pseudo operation 529
- dw, pseudo operation 54
- dynamic RAM
  - cell design 389
  - chip organization 390
  - nibble mode 394
  - page mode 394
  - precharging 388
  - refresh 389, 393
  - static column mode 394

## E

- EBCDIC code 10
- Eckert, Presper 21
- economics in computer design 84
- edge-triggered flip-flops, using 170
- EDSAC 21
- EDVAC 21
- EEPROM 394
- effective address 45
  - MC68000 97
  - SRC 58
- effective address, defined 446
- emulators
  - used to measure system performance 15
- encoding instructions 32
- Endians 377
- ENIAC 20
- EPROM 396
- EQU, assembler pseudo operation 529
- equ, pseudo operation 54
- error detection and correction 498–503
  - CRC 502
  - Hamming codes 499
  - in communications systems 575

- parity checks 498
  - SECDED coding 498
  - Ethernet 574, 584–587
    - addresses 587
    - cabling (Table) 585
    - Collision detection in 586
    - data link layer 586
    - interface to other networks 587
    - MAC addresses 587
    - packet structure 587
    - physical layer 585
  - exception
    - vector-defined 114
  - exceptions 9, 183–191
    - alignment 186
    - arithmetic errors 187
    - data and instruction access 186
    - that halt instructions,
      - complications of 190
    - hardware-caused 187
    - interrupts as external 187
    - kinds 186–188
    - machine check 186
    - MC68000 114
    - non-maskable 115, 187
    - in pipelined designs 218
    - privileged instruction 187
    - process of 184–186
    - program generated 187
    - in SRC 188–190
    - system reset 186
    - trace and debugging 187
    - unimplemented instruction 187
  - vectors
    - MC68000 114
  - expression evaluation
    - with 3- 2- 1- and 0-address machines 43
- ## F
- Fairchild Semiconductors 22
  - FDM, and Frequency-Division Multiplexing 571
  - fetch-execute cycle 6
  - FireWire 13, 587–589
  - fixed point numbers 314
    - addition 322
  - flat panel displays 557
  - flip-flops
    - edge-triggered 70
    - edge-triggered, in processor design 170
    - hold time 70
    - level-sensitive, in processor design 170

- master-slave 70
  - setup time 70
- floating point hardware 362–370
- floating point numbers
  - addition and subtraction 366
  - IEEE floating point format 365
  - multiplication and division 368
  - normalization 364
  - representation of 363
- flops (floating point operations per second) 87
- Flynn limit 280
- formal and informal machine descriptions 63
- forwarding, of data in pipeline designs 239
- four address machines 38
- frame
  - communications packet 573
- frames
  - RS-232 serial 580
- front panel 21
- full-duplex communications 571

## G

- G (giga), defined 2
- gain error—in analog conversion 565
- gate signals
  - estimating minimum 168
- general purpose registers 32
- general register machines 32, 43
- giga, defined 2
- goto statement
  - infinitely abusable 111
- Gulliver's Travels 379

## H

- half-duplex communications 571
- Hamming codes 499
  - error syndrome 500
- Hansel and Gretel 112
- hardwired control unit 171
- hazards, pipeline 256–279
- higher level languages
  - mapping to assembly language 11
- hit ratio, in two level hierarchy 421
- Honeywell 21
- hubs, network 574

## I

- I/O
  - asynchronous 480
  - bus structures 475

- Centronics interface 485
  - character 479
  - device drivers 483
  - DMA 494–497
  - drivers 481
  - interrupt driver 489
  - interrupts 486
  - interrupts, hardware for 488
  - isolated 116
  - memory mapped 116, 476
  - processors 497
  - programmed, hardware for 477
  - programming for 481
  - subsystems, requirements for 474
- IBM
- IBM801 9
  - System/364, 23
- IBM 75GXP 547
- IEEE floating point format 10, 364
- IEEE1394 (FireWire) 14, 587–589
- immediate addressing 45, 65
- implementation domain
- defined 17
  - importance of 18
- indexed addressing 47, 66
- indirect addressing 46, 66
- ink-jet printers 558
- input device drivers 483
- input devices
- keyboards 560
  - mouse 560
- input/output
- see I/O 474
- instruction
- encoding 32
- instruction classes 9
- instruction prefetch 92
- instruction register 6
- instruction set
- impact on pipeline design 240
- instruction set architectures 6
- instruction sets 32, 33
- ALU instructions 36
  - branch instructions 37
  - data movement instructions 35
  - as tower of Babel 33
- instructions
- issuing 11
  - pipelining 90
  - prefetching 90
  - relocatable 50
- Intel 23
- 8086
    - memory 379
    - memory segmentation 451
    - register structure 7
  - 8088, memory 379
  - P6, superscalar processing 281
  - Pentium 4, register structure 8
  - Pentium cache 429
- interfaces, analog
- See analog interfaces 561
- Internet, the 591
- applications 601
  - ARPA 569
  - ARPANET 569
  - CIDR domain routing 597
  - DARPA 569
  - DHCP 598
  - DNS 591
  - domain names 591
  - domains, top-level 595
  - future directions 600
  - ICANN 595
  - IP addresses 592, 595–599
    - Class A, B, and C 597
    - who assigns 595
    - in wristwatches 601
  - IP level 593
  - IPv4 592
  - IPv6 601
  - names 591
  - NAT 598
  - network masks 599
  - packet reception 595
  - packet routing in 593
  - routers in 593
  - subnet masks 599
  - subnets 598
  - TCP level 593
- internetworking 574
- interrupt service routine
- MC68000 114
- interrupts
- See also exceptions
  - external exceptions as 187
  - hardware interface 487
  - I/O 486
  - I/O drivers 489
  - MC68000 114
  - nested 492
  - effect on pipeline 239
  - priorities
    - MC68000 115
  - priority 488

- IP addresses
    - wasted IP addresses 597
  - IPv4 592
  - IPv6 601
  - ISA 7
  - ISO, International Standards Organization 570
  - isochronous data transmission 588
  - isolated I/O 116
- J**
- JEDEC 392
- K**
- K (kilo), defined 2
  - Kernighan, Brian 111
  - keyboards 560
  - Kilby 22
  - kilo, defined 2
- L**
- LANs 572, 584–587
    - bridges in 574
    - routers 575
  - laser printers 558
  - latch
    - triggering mode 70
  - latency 23
  - layer models
    - exceptions to 592
    - TCP/IP vs. OSI 591
  - level-sensitive flip-flops, using 170
  - linkers 524
  - load-store machines 44
  - locality, temporal and spatial 419
  - location counter, assembler 533
  - logic circuits
  - logic gates
    - data transmission view of 68
    - propagation time 70
- Lotus 23
- M**
- M (mega), defined 2
  - μ (micro), defined 2
  - m, defined 2
  - MAC (Media Access Control) addresses, in networks 587
  - machine interrupts—see interrupts
- machine reset 181–183
    - MC68000 181
    - possible actions 181
    - signal, sources of 182
  - machine state, defined 9
  - magnetic tape 549
  - main memory
    - boards 397–416
    - interleaving 405
    - modules 397–416
    - RAM cells and chips 381
    - RAM vs. ROM 380
    - read and write operations 380
  - Manchester encoding 572
  - mapping
    - higher level language to assembly language 11
  - matrix decoders 386
  - Mauchly, John 21
  - MC6800 93
  - MC68000 5, 93–117
    - addressing modes 97
      - absolute 102
      - address register indirect 100
      - autoincrement and autodecrement 100
      - based 100
      - based indexed 101
      - direct 103
      - immediate 103
      - relative 102
      - relative indexed 102
    - effective address calculation 99
    - example programs
      - clear a block of words 113
      - I/O: reading from the keyboard 117
    - exception processing
      - RTN description 115
    - exception vector 114
    - exceptions 114
    - history of 89
    - I/O 116
    - instruction interpretation 105
    - instruction set 104
      - arithmetic and logic instructions 107
      - bit operations 108
      - branch instructions 110
      - program control instructions 110
      - shift and rotate instructions 109
    - interrupts 114
    - machine reset 181
    - operands, value of 103



- privileged instructions 105
- processor state
  - RTN description 96
- traps and traces 114
- word size 94
- M6809 93
- mega, defined 2
- memory
  - cache 422–444
  - EEPROM 396
  - EPROM 396
  - hierarchy 419
  - Intel 8086 379
  - Intel 8088 379
  - performance parameters 381
  - PowerPC G4 379
  - PROM 396
  - ROM 396
  - system 465
    - block size selection in 465
    - disk cache in 466
    - instruction and data accesses 465
- memory access time 381
- memory bandwidth 381
- memory boards
  - 2-D chip arrays 400
  - expanding of words 400
- memory boards and modules
- memory cell 382
- memory chips 397
- memory hierarchy
  - I/O influence on 466
  - multiprogramming in 421
  - impact of technology on 467
  - two level 416–421
    - address translation 436
    - cache regime in 420
    - demand paging 447
    - disk regime in 421
    - hit and miss ratio 417
- memory latency 381
- memory management
  - paging 448
  - unit
    - in virtual memory 444
- memory mapped I/O 116
  - in MC68000 116
- memory mapped video 553–557
- DMA
  - use with memory modules 411
- memory modules 402–416
  - DMA used with 411
  - interleaving 411
- memory state, defined 9
- memory system (Figure) 14
- memory system design 377–467
- memory-CPU interface 378
- memory-memory machines 44
- messages, undesired
  - bus error 448
  - segmentation fault 448
- mflops (millions of floating point operations per second) 87
- microcoding 191
- microprogramming
  - bit ORing 203
  - branching and looping in 195
  - control store 194
  - control unit 191, 195
  - general approach 192
  - horizontal and vertical 198
  - nanocoding 203
  - SRC example 194
  - SRC implementation 199–203
  - subroutines in 204
  - writable control store 203
- Microsoft 23
- milli, defined 2
- mips
  - as performance measure 87
- MIPS R2000, branching conditions in 358
- miss ratio
  - in two level memory hierarchy 417
- modem, null 580
- modems 581–582
- monotonicity—in analog conversion 561
- Moore School of Engineering 21
- Motorola
  - MC6800, see MC6800
  - MC68000, see MC68000
  - PPC 601, see PPC 601
- mouse 558
- multilevel page table 450
- Multiple instruction issue
  - defined 211
- multiplication, integer
  - algorithms and hardware 338–346
  - Booth recoding 346
  - parallel hardware 342
- multiprogramming 421

mutiplication, floating point 365  
 mysteries, no xiii

## N

- n, defined 2
- nano, defined 2
- nanocoding 203
- NAT 598
- nested interrupts 492
- Network Address Translation 598
- network layer, in communications 577
- networking
  - bridges 569
- networks
  - addresses, IP 587
  - application layer 577
  - circuit switching 572
  - communications
    - full-duplex 571
    - half-duplex 571
    - simplex 571
  - data link layer 577
  - Ethernet 574
  - hubs 585
  - internetworking 574
  - LAN topology 573
  - network layer 577
  - OSI layer model 577–578
  - packet switching 572
  - physical layer 577
  - presentation layer 577
  - protocols
    - TCP/IP 591–593
  - routers 574–575
  - session layer 577
  - store-and-forward 573
  - structures 570–571
  - TCP/IP 591–593
  - Token Ring 574
- topology
  - bus 573
  - Ethernet 574
  - ring 573
  - star 573
  - star-bus 574
  - star-ring 574
  - Token Ring 574
- transport layer 577
- virtual circuits 573
- nibble mode RAM 557
- nibble, defined 2
- normalization, floating point numbers 366
- Noyce, John 22
- NRZ signal encoding 572
- NRZI signal encoding 572
- null modem 580
- number systems 308
  - complement 316–320
- numbers
  - fixed point 314
- nybble, defined 2

## O

- of 76
- offset error—in analog conversion 565
- one-address (accumulator) machines 41
- open collector logic 72
- operand access 36
- ORG, assembler pseudo operation 530
- org, pseudo operation 54
- OSI layer model, in communications 577, 578
- overflow, calculation of in ALU 357

## P

- packet switching, networks 572
- packets, communications 572
- page fault 191
- page mode RAM 394
- paging
  - in virtual memory 421
- parallel buses 12
- parallel interface, Centronics 485
- Parallelism
  - instruction-level 280–303
- parity checking 498
- Patterson, David 9
- PCI bus 14
- peak bandwidth 475
- Pentium 35
- performance measurement
  - calculating speedup 85
  - Dhrystones 88
  - flops, mflops 87
  - mips 87
  - spec 88
  - synthetic benchmarks 87
  - Whetstones 87
- peripheral devices 15, 537–557
  - analog interfaces 537–540
  - disk drives 537–547

- display devices 550–558
- input devices 560
- printers 558–560
- physical address, defined 446
- physical layer, in communications 577
- pipeline
  - registers 240
  - WAR and WAW hazards 258
- pipelining 24, 90, 212
  - ALU instructions 244
  - basic assumptions in design of 239
  - branch delay slots in 254
  - branch instructions in 246
  - classifying instruction set for 241
  - control signal generation 250
  - control signals for 242
  - data forwarding in 261, 265
  - data path design 246
  - dependence in 241
  - design technique 241
  - exceptions in 279
  - global and local states in 246
  - hardware requirements, additional 240
  - hazard detection by compiler 263
  - hazard detection by hardware 263
  - hazards in 256–264
  - hazards, branch 263
  - hazards, data 257
  - important notice about 244
  - in SPARC 129
  - impact of instruction set on 241
  - load and store instructions 244
  - memory requirements 240
  - overview of 212
  - register file structure in 240
  - SRC design 241–265
  - stalls 263
- platters
  - disk drives 538
- positional notation 312
- Postscript 559
- PowerPC G4
  - ftiwz instruction, complicated in anyone's book 93
  - memory 379
- PPC G4 7
  - cache design 441
- PPC601
  - virtual memory 462
- prefetching 90, 92
- presentation layer, in communications 577
- printers 558–560
  - dot-matrix 558
  - ink-jet 560
  - interfaces 558
  - laser printers 560
  - Postscript 560
- procedure calls 9
- processor
  - design
    - control sequences 161
    - design technique 148
    - design, control unit 167
    - gate-level design of 159
    - multiple bus designs 175
    - state, defined 9
    - status word 37
  - program counter
    - definition 6
    - gate level design 19
  - programmed I/O, hardware for 477
  - programmer's
    - manual 9
    - model 7
  - PROM 396
  - propagation delay 169
  - propagation time 70
  - protocols, communications 570
  - pseudo operations
    - dw 54
    - equ 54
    - org 54
  - pseudo operations, in assemblers 529
  - pull-up resistor 72
- R**
- radix
  - complement 318
  - complement, diminished 318
  - conversion 310
- radix conversion 313
- RAID disk arrays 547, 547
- RAM
  - cells and chips 381
  - chip design 381
  - cost of 385
  - DDR (double data rate) SDRAM 393
  - decoders, tree and matrix 386
  - dynamic (DRAM) 392
  - dynamic 390
  - nibble mode 394
  - page mode 394
  - static 382
  - static column mode 394
  - static, cell design 387
  - synchronous (SDRAM) defined 392
  - VRAM 394
- raster 552
- RCA 21
- refresh 394
- refresh, dynamic RAM 390
- register 5

- register addressing 66
- register indirect addressing 46, 66
- register transfer languages 16
- register transfer notation—see RTN
- register transfers
  - using edge-triggered flip-flops 170
  - gate level design of 19, 66–78
  - using level-sensitive flip-flops 170
  - strobe signals 67
- register windows 118
- register-memory machines 44
- register-to-register machines 44
- relative addressing 47, 66
- relay computer 20
- reset. See machine reset 181
- ring network topology 573
- ripple-carry adders 324, 651
- RISC
  - vs. CISC 9
  - compiler's role in 92
  - design philosophy 90
  - philosophy 8
  - vs. CISC 88
- Ritchie, Dennis 111
- ROM 395
- rotate, hardware for 360
- routers
  - Internet 587
  - LAN 570
  - network 570
- RS-232 data communications 569–570
  - asynch. frames 579
  - data communications equipment 578
  - data link layer 577
  - data terminal equipment 578
  - physical layer 577
  - session layer 577
  - signals and pins 578
- RTN 15, 55
  - abstract vs. concrete descriptions 77, 148
  - description of addressing modes 64
  - description of MC68000 96
  - use in pipelining 246
  - SRC description 55–64, 687–690
  - symbols used in
- Rules of buses 75
- S**
- SDRAM 390
- SECEDED (single error correct, double error detect)
  - coding 499
- semantic gap 89
- serial buses 13
- session layer, in communications 577
- Shannon, Claude 20
- shift hardware 360
  - barrel shifter 360
- Shockley, William 22
- signal dispersion 169
- sign-magnitude number representation 317
- simplex communications 571
- skew
  - clock 169
- SMART disk drive technology 547
- SPARC 117–132
  - addressing modes 123
  - advanced implementations 131
  - architectural overview 117
  - background 117
  - instructions 122, 124
    - arithmetic 125
    - branch and program control 127
    - data movement 105
    - logical and shift 126
  - operand types 122
  - pipelining in 130
  - programmer's model 118
  - register windows 120
  - registers 120
  - tagged data type 122
- spatial locality 418
- spec (performance measurement) 88
- speculative execution 92
- speedup, calculating 85
- SPOC 24
- SRC
  - 1-bus control signals
    - ADD, SUB, AND, OR, SHR, SHRA, SHL, SHC, NOT, NEG, C = B, INC 161
    - A<sub>in</sub> 160
    - BA<sub>out</sub> 157
    - c1<sub>out</sub>, c2<sub>out</sub> 159
    - C<sub>in</sub>, C<sub>out</sub> 160
    - CON 166
    - CON<sub>in</sub> 166
    - CountIn 173
    - Decr 165
    - End 163
    - Goto6 165
    - Gra, Grb, Grc 157
    - IR<sub>in</sub> 156
    - Ld 164
    - Load 173
    - MA<sub>in</sub> 159
    - MD<sub>bus</sub> 159
    - MD<sub>rd</sub> 159
    - MD<sub>wr</sub> 159

- n=0 164
- PC<sub>in</sub>, PC<sub>out</sub> 156
- R<sub>in</sub>, R<sub>out</sub> 157
- 1-bus design 149–175
  - see also SRC 1-bus design 149–175
- abstract RTN description 55–64, 148
- addressing modes
  - informal description 47
- arithmetic and logic instructions
  - formal description 63
  - informal description 63
- branch instructions
  - formal description 63
  - informal description 52
- concrete vs. abstract RTN descriptions 148
- condition codes, calculation of 357
- effective address 58
- exception processing 188–190
- I/O 62
- informal description 47–54
- instruction execution 60
- instruction formats
  - formal description 57
  - informal description 47
- instruction interpretation 59
- instructions
  - add 51
  - addi 51
  - and 51
  - andi 51
  - br 52
  - brl 52
  - edi 189
  - een 189
  - la 48
  - lar 48
  - ld 48
  - ldr 48
  - neg 51
  - not 51
  - or 51
  - ori 51
  - rfl 189
  - ri 189
  - shifts 52
  - st 48
  - str 48
  - sub 51
  - svi 189
- load and store instructions
  - formal description 60
  - informal description 47
- machine reset
  - abstract RTN for 182
  - concrete RTN for 183
- microprogrammed implementation 192, 199–204
- pipelined design 240–266
- programmer's model 47
- registers and memory
  - formal description 55
  - informal description 47
- RTN description
  - arithmetic instructions 61
  - branch instructions 60
  - disp 58
  - instruction formats 57
  - instruction\_execution 60
  - instruction\_interpretation 59
  - load and store instructions 60
  - main memory 57
  - nop 62
  - processor state 56
  - rel 58
  - stop 62
- VLIW implementation 290
- SRC 1-bus design
  - clocking and timing 167
  - concrete RTN
    - add instruction 150
    - addi instruction 152
    - br instruction 153
    - ld instruction 152
    - shift instructions 154
    - st instruction 152
  - control sequences 162
    - add instruction 162
    - addi instruction 163
    - branch instructions 165
    - ld instruction 163
    - shift instructions 164
  - control unit 167
    - system initialization 174
  - control unit, hardwired 171
  - data path design 154
  - gate-level design
    - ALU 160
    - instruction register 158
    - MA and MD register design 159
    - memory interface 159
    - register file 157
    - temporary registers 160
  - hardwired control unit 171
- SRC 2-bus design 175
  - add instruction 175
  - control signals, Sra, Srb, and Src 175
- SRC 3-bus design 178–179
  - add instruction 178
  - control signals, GA<sub>rc</sub>, RA<sub>out</sub>, GB<sub>rb</sub>, RB<sub>out</sub> 179
- stack machines 32
- stalls, pipeline 263

- star network topology 574
  - star-bus network topology 574
  - star-ring network topology 574
  - static column mode RAM 394
  - static RAM
    - cell design 387
    - timing 388
  - Stibitz, George 20
  - store-and-forward networks 573
  - stored program concept 6
  - stored program optical computer, SPOC 24
  - strobe signal 67
  - strobe signals
    - minimum 169
  - subnets, in Internet 598
  - superscalar operation 90
  - Swift, Jonathan 379
  - switching fabrics 360
  - symbol table, in assembler 533
  - symbols, RTN
  - syndrome, of Hamming encoded word 500
  - system initialization 174
- T**
- T (tera), defined 2
  - TCP/IP network protocol suite 591–593
  - TDM, time-domain multiplexing 571
  - temporal locality 418
  - tera, defined 2
  - Texas Instruments 23
  - three-address machines 40
  - time-division multiplexing 571
  - timing 167
    - control signals 76
    - estimating minimum clock period 168
    - estimating minimum gate signals 169
    - gate signal 70
    - and register transfers 66–78
    - strobe signal 67
  - timing parameters
    - estimating for implementation domain 169
  - TLB-translation lookaside buffer 453
  - Token Ring networks 574
  - tower of Babel 33
  - traces
    - MC68000 114
  - transfer descriptor, USB 579
  - transport layer, in communications 567
  - traps
    - MC68000 114
  - tree decoders 440
- tri-state gates and logic 73
- Turing
  - Alan 2
  - machine 2
- two address machines 40
- two pass assemblers 534
- U**
- UNIVAC 21
  - unsigned addition 323
  - unsigned integer
    - value of 312
  - upward compatibility
    - at binary level 85
    - binary with exceptions 85
    - by emulation 85
    - at source code level 85
  - USB 587–591
    - defined 14
    - transfer descriptor 589
  - UTP, Unshielded Twisted Pair 585
- V**
- VAX11 7
    - pseudo-operations 529
  - vector instructions
    - defined 281
  - vector processors 281
  - vector, exception-defined 114
  - video display terminals 553–557
    - character generators in 555
  - video monitors 550–553
    - bandwidth and resolution 553
    - color 552
  - video RAM 394
  - video, memory mapped 553, 554
  - views of the computer 3
    - logic designer's 16
    - machine/assembly language programmer's 5
    - user's 5
  - virtual address, defined 446
  - virtual circuits, in networks 573
  - virtual memory 444–465
    - address translation in 462
    - defined 444
    - demand paging 447
    - disk regime 421
    - effective address, defined 446
    - logical address, defined 446

memory management  
  paging 447  
  unit 447  
multiprogramming 421  
page fault 449  
page placement and replacement 454  
page tables 450  
physical address, defined 446  
PPC601 460  
primary and secondary levels,  
  defined 454  
as a system 454  
TLB-translation lookaside buffer 453  
virtual address, defined 446

VLIW

  architectures 211, 281

Von Neumann machine 21

Von Neumann, John 21

VRAM 394

**W**

wave pipelining 24, 76  
Weiner, Norbert 20  
Whetstones (performance measurement) 87  
Wilkes, Maurice 24  
wired AND 72  
wired OR 72  
word size 94  
  memory vs. CPU 376  
word, defined 2  
writable control store 203  
write after read (WAR) hazards 257  
write after write (WAW) hazards 257  
write back cache memory policy 466  
write through cache memory policy 466

**Z**

Zuse, Konrad 20



Computer Systems  
Design and Architecture

# Computer Systems Design and Architecture

Second Edition

Vincent P. Heuring | Harry F. Jordan | T. G. Venkatesh

*Computer Systems Design and Architecture 2e* places emphasis on issues related to both architecture as well as organization of the computer. It interrelates three different viewpoints to provide unique understanding of the subject: the perspectives of the logic designer, the assembly language programmer and the computer architect. The text describes both CISC and RISC models at the ISA level using the formal description language of RTN (Register Transfer Notation), allowing for an in-depth appreciation of different machine structures and functions.

## Salient Features

- ❖ In-depth coverage of architectural trends such as pipelining, superscalar technique, and VLIW
- ❖ Discusses the OSI layer model and its relationship to TCP/IP Internet Protocol
- ❖ Provides thorough understanding of static and dynamic RAM design, cache memory management, virtual memory and arithmetic unit
- ❖ Detailed discussion on design of datapath and pipeline of 1, 2 and 3 bus processors
- ❖ Concepts are supplemented with practical design issues in modern processors such as Pentium, SPARC, ARM and Power PC

This edition is manufactured in India and is authorized for sale only in India, Bangladesh, Bhutan, Pakistan, Nepal, Sri Lanka and the Maldives.

ISBN 978-81-775-8483-7



www.pearsoned.co.in