

Computer Architectures and Operating Systems (CAOS)

Study Notes

Dr Chris Crispin-Bailey

Revised Summer 2022

Disclaimer: All trademarks used herein are property of their respective owners, and are neither endorsed or subject to any claim of ownership by the author or publisher. Microsoft, Windows, MS-DOS, Apple, Mojave, Android, Chrome, INTEL, ARM, AMD, ASUS, Fujitsu, Cray, IBM, Google, Samsung, Hisilicon, Huawei, Qualcomm, Transcend, Crucial, Infineon, and Seagate, are registered trademarks of their respective owners.

Use of these marks does not imply any affiliation or endorsement between any of these entities and the Course provider. Wherever possible, such marks have been capitalised to identify them. However, the reader should make their own enquiries regarding the status of such instances of brand names and trade-marks that may appear within this text, capitalised or otherwise.

The use of imaginary companies and references may be used occasionally in this text for the purposes of illustration and embellishment of case-study material. Any similarity to actual companies past or present is entirely coincidental. Though information in this text has been carefully checked and is believed correct at the time of publication. No liability is accepted, and no warranty implied or given for any information contained within this document.

Copyright: This material is subject to copyright. Any unauthorised reproduction of content in any form, or storage in a retrieval system, electronic or otherwise, is prohibited without express permission of the author.

The University of York, UK, has express permission to scan and digitally reproduce this material for teaching of students registered on undergraduate and postgraduate degree programmes at the University of York.

Acknowledgments: This document was typeset, in Lato and Lucida Sans typeface, with the help of KOMA-Script^a and Latex^b using the kaobook class^c.

^a KOMAScript: <https://sourceforge.net/projects/koma-script/>

^b LATEX: <https://www.latex-project.org/>

^c Kaobook: <https://github.com/fmarotta/kaobook/>

^d Manuscript Ref. 2.05

Detailed Contents

Detailed Contents	iii
1 Introduction	1
1.1 Introduction	1
1.1.1 Important Notes	1
2 The Fundamentals	3
2.1 Some initial observations	4
2.2 Important units of measurement and specification	5
2.2.1 Kilo, Mega, Giga, and more	6
2.3 A conceptual computer system	7
2.3.1 von Neumann model	8
2.3.2 The von Neumann Bottleneck	9
2.4 The software viewpoint	10
2.5 Putting electronics in the picture	12
2.5.1 Speed versus complexity	14
2.6 Summary	16
2.7 Terminology used in this chapter	17
3 The Heart of The Machine	19
3.1 Processors: past, present, and future	20
3.2 So what does a CPU do?	21
3.2.1 Heat and power	22
3.2.2 What exactly is computation?	24
3.2.3 Toward a better definition	25
3.3 The performance barrier	26
3.3.1 Processor frequency	33
3.3.2 The power of multiples	34

3.3.3	Guesswork: prediction and speculation.	35
3.3.4	Branch prediction	36
3.3.5	Speculative execution	38
3.4	CPU internal architecture	38
3.4.1	An instruction 'in flight'	41
3.4.2	A more advanced microarchitecture	43
3.5	A little bit of programming	44
3.6	Extending the Instruction Set	47
3.7	More on instruction microsequences	48
3.8	Summary	49
3.9	Terminology introduced in this chapter	50
4	Computers Everywhere	51
4.1	A machine with many facets	52
4.2	The general purpose computer	52
4.3	General purpose computers	53
4.3.1	The computer in your hand.	54
4.3.2	The hidden computer	55
4.4	Being specific	56
4.5	The impact on processor technology.	58
4.6	Mainframes and supercomputers	58
4.7	The internet of things	60
4.8	Portable, wearable, implantable	61
4.9	Parallel processing	62
4.9.1	Speedup vs Amdahl's Law	63
4.10	Summary	65
4.11	Terminology	66
5	Making Memories	67
5.1	Memories, past and present	68
5.1.1	Getting physical	70
5.2	Non-volatile memories	71
5.3	Volatile memories	73
5.3.1	Static Random Access Memory (SRAM)	74
5.3.2	Dynamic Random Access Memory (DRAM)	74
5.4	Memory performance strategies	75
5.4.1	Access time and cycle time	76
5.4.2	Memory access protocol	76
5.4.3	Memory width	77

5.4.4	Single-cycle versus Burst-mode access	77
5.4.5	Page mode access	80
5.5	Cache memory	82
5.5.1	Multilevel cache	85
5.6	Cache management policies	87
5.6.1	Mapping policies	87
5.6.2	Replacement policies	89
5.6.3	Write-back policies	90
5.7	Avoiding errors	92
5.8	Programming considerations	93
5.9	The future of memory devices	95
5.10	Summary	97
5.11	Terminology introduced in this chapter	99
6	Building The System	101
6.1	Putting it all together	102
6.2	The system bus vs the dedicated bus	104
6.2.1	Concurrency in bus architecture	106
6.3	Bus standards	107
6.4	A generic system bus	108
6.5	PCI: a very successful Bus Standard	110
6.6	Data rate matching and buffers	113
6.7	USB: Universal Serial Bus	113
6.7.1	Plug-and-Play	116
6.7.2	Hubs and ports	116
6.8	More common bus standards	116
6.9	Industrial and embedded standards	117
6.9.1	I2C (Inter-Integrated Chip) bus standard.	118
6.9.2	CAN bus	120
6.10	Rack-mount, hot-swap, and servers	121
6.11	IO device mapping and IO servicing	122
6.12	Summary	123
6.13	Terminology introduced in this chapter	125
7	Data Storage	127
7.1	Storing Data	128
7.2	Disks and data storage	128
7.2.1	From tapes to disks	128
7.2.2	Modern disks in detail	131

7.2.3	A disk performance example.	134
7.2.4	The cost of fragmentation	137
7.3	Disk cache	138
7.4	Solid State Drives (SSD)	139
7.4.1	SSD memory wear	141
7.4.2	Storage density	142
7.4.3	Buyer beware	143
7.5	Storage reliability and fault tolerance	144
7.6	Remote storage	145
7.7	Summary	146
7.8	Terminology introduced in this chapter	148
8	A World of Peripherals	149
8.1	Everyday peripherals	150
8.1.1	Input devices	150
8.1.2	Output devices.	152
8.2	Novel peripheral interfaces	155
8.3	Networking connectivity	156
8.3.1	Wired networks	156
8.3.2	Wireless networks	158
8.3.3	Network protocols and overheads	159
8.4	Summary	162
8.5	Terminology introduced in this chapter	163
9	Operating Systems Concepts	165
9.1	The idea of an Operating System	166
9.1.1	A 'bare-metal' system.	166
9.1.2	BIOS	167
9.1.3	Low-level input/output drivers	167
9.2	The operating system hierarchy	168
9.3	Choices, choices	170
9.4	The purpose of the kernel	172
9.4.1	Abstraction	173
9.5	The OS as taskmaster	174
9.6	General vs specific, once again	176
9.7	Summary	178
9.8	Terminology introduced in this chapter	179

10 Workload Management	181
10.1 The task in hand	182
10.2 Process ID and housekeeping	183
10.3 Thread management models	184
10.4 Task scheduling	185
10.5 Scheduling algorithms	186
10.5.1 Task scheduling queues	188
10.6 Pre-emptive scheduling	188
10.6.1 Multiple queues	190
10.6.2 Interrupts and polling	191
10.7 Achieving priority	193
10.8 Advanced CPU architecture OS support	194
10.8.1 From scalar to superscalar	194
10.8.2 Multicore	195
10.8.3 Hyperthreading and SMT	196
10.9 Summary	199
10.10 Terminology introduced in this chapter	200
11 Memory Management	201
11.1 Memory management	202
11.2 Memory allocation	202
11.2.1 Memory access control	204
11.3 Virtual memory	205
11.3.1 Paging cost penalties	208
11.4 Inter-Process Communication	209
11.4.1 Pipes	209
11.4.2 Message queues	210
11.4.3 Shared memory	212
11.5 Privileges and restrictions	212
11.6 Summary	216
11.7 Terminology introduced in this chapter	217
12 File System Management	219
12.1 Foreword	220
12.2 The concept of a file system	220
12.3 A file-system example	222
12.4 The boot sector	222
12.5 Volumes and partitions	223
12.6 The root directory	224

12.7	File-system hierarchy	224
12.8	Special file cases	225
12.9	File attributes	227
12.10	The variety of file systems	229
12.11	File compression	230
12.11.1	File compression example	230
12.12	Automatically compressed file systems	232
12.12.1	File encryption	233
12.13	File-system resilience	234
12.14	Internal file formats	235
12.14.1	File structures and encodings	236
12.14.2	File format tradeoffs: an example	239
12.15	File management	241
12.16	Summary	243
12.17	Terminology introduced in this chapter	244

13 Networks: Being Connected 245

13.1	Foreword	246
13.2	Putting modern networks in context	246
13.3	The OSI reference model	247
13.3.1	TCP/IP: Another example of layer abstraction	249
13.3.2	Security	250
13.4	Network structures	250
13.4.1	Bridging the gap	251
13.4.2	Switching things around	253
13.5	Routing	254
13.6	Network node identification	256
13.7	Network services	259
13.7.1	Sockets Interface	260
13.8	Distributed systems	261
13.9	A web-based client-server system	262
13.10	Cloud computing	265
13.10.1	Traditional in-house infrastructure	265
13.10.2	The cloud computing option	266
13.11	Compute servers and virtual machines	267
13.12	Virtualisation	269
13.13	Summary	272
13.14	Terminology introduced in this chapter	273

14 Keeping It Safe	275
14.1 Foreword	276
14.2 Security threats	276
14.3 Viruses	277
14.3.1 Virus checkers	277
14.3.2 Virus examples	278
14.4 Architectural exploits	279
14.5 Firewalls	280
14.6 Encryption and validation	281
14.7 Data Resilience	282
14.8 Resilient systems	285
14.9 UPS systems	287
14.10 Resilience and safety	288
14.11 Summary	289
14.12 Terminology Introduced in this chapter	290
 Appendices	 291
A Glossary of Terminology	293
A.1 Glossary of terms used in this book	293
 Alphabetical Index	 317

Figures

1.1	Electronic Radio Valve	1
2.1	Simple Computer System Block Diagram	8
2.2	Standard Logic Gates, symbols, and input/output 'truth tables'. .	13
2.3	Simple examples of Speed and Complexity	14
3.1	Electronic components, ancient and modern	21
3.2	A processor with heat sink	23
3.3	Simplified microarchitecture of a CPU	39
3.4	A typical micro-instruction sequence	41
3.5	Non-pipelined Instruction Microsequence	42
3.6	Pipelined Instruction Microsequence	43
3.7	A Superscalar Microarchitecture	44
3.8	3-Way Multiport Register File	44
3.9	Variations of instruction microsequences.	48
4.1	CPU Examples	55
4.2	NAT Miniature EEG sensor	56
4.3	Supercomputers	60
5.1	Core-store Memory Modules	69
5.2	SIMM and DIMM memory modules	70
5.3	DRAM single read operation	78
5.4	DRAM burst read operation	79
5.5	DRAM fast-page-mode read operation	81
5.6	A scalar microarchitecture with dual on-chip cache hierarchy . .	86
5.7	Cache Mapping Schemes	89
5.8	Memory Map Example	94
5.9	DRAM memory module capacity trends	96
6.1	Motherboard example	103
6.2	System diagram with more comprehensive bus architecture . . .	105
6.3	A simple hypothetical bus example	108
6.4	Data transfer efficiency on PCI bus	112
6.5	Some additional commonly used bus standards	117

6.6	I2C bus system example	119
6.7	Differential signalling concept	121
7.1	Punched card Program Instruction	128
7.2	Tape System Conceptual operation	129
7.3	Drum Storage concept	130
7.4	Disk unit concept	132
7.5	Modern HDD and SSD units compared	143
8.1	A system with typical Peripherals	151
8.2	Golf Ball Printer	154
8.3	Network Efficiency and Latency	161
9.1	Operating System Hierarchy	169
9.2	Examples of OS types	172
9.3	A desktop OS Process monitoring tool	176
10.1	CPU workload process Monitor	183
10.2	Thread management models	185
10.3	SMT example	198
11.1	Virtual memory page Swapping	207
11.2	Memory usage versus physical memory	208
12.1	Volumes and partitions Example	224
12.2	Example hierarchical file system	226
12.3	Linked files example	227
12.4	Example of file attributes	229
13.1	OSI reference model	248
13.2	TCP/IP model	249
13.3	Simple network example	250
13.4	Bridged network example	252
13.5	Switched network example	254
13.6	Remote central heating network application	263
13.7	Client-Server Heatweb application	264
13.8	Typical virtualisation models	271
14.1	Parity checking Examples	284
14.2	Gaussian MTBF curve	285