

---

# OPERATING SYSTEMS: CONCEPTS AND DESIGN

---

Milan Milenković

Technische Hochschule Darmstadt  
FACHBEREICH INFORMATIK

**BIBLIOTHEK**

Inventar-Nr.: ..... 20149 .....

Sachgebiete: ..... 2.5 .....

Standort: ..... 1987 .....



**McGraw-Hill Book Company**

New York St. Louis San Francisco Auckland Bogotá Hamburg  
London Madrid Mexico Milan Montreal  
New Delhi Panama Paris São Paulo Sydney Tokyo Toronto

---

# CONTENTS

---

Preface	xv
<b>1 INTRODUCTION</b>	<b>1</b>
1.1 Evolution of Operating Systems	2
Serial Processing	2
Batch Processing	4
Multiprogramming	6
1.2 Types of Operating Systems	8
Batch Operating Systems	8
Multiprogramming Operating Systems	9
Time-Sharing Systems	10
Real-Time Systems	11
Combination Operating Systems	12
1.3 Different Views of the Operating Systems	12
The Command-Language User's View of the Operating System	13
The System-Call User's View of the Operating System	14
1.4 Design and Implementation of Operating Systems	15
Functional Requirements	16
Implementation	18
1.5 Summary	20
Other Reading	21
<b>2 INPUT/OUTPUT: PRINCIPLES AND PROGRAMMING</b>	<b>22</b>
2.1 The Input/Output Problem	23
Asynchronous Operation	23
The Speed Gap: CPU versus Peripherals	24

	Format Conversions	25
2.2	Input/Output Interfaces	26
	Buffer Registers	30
	Command Registers	31
	Status Registers	32
2.3	I/O Port Examples	32
	Universal Synchronous/Asynchronous Receiver/Transmitter (USART)	32
	Programmable Interval Timer (PIT)	36
2.4	Program-Controlled I/O	37
	Controlling a Single Device	37
	Controlling Multiple Devices: Polling	44
2.5	Interrupt-Driven I/O	46
	Controlling a Single Device	47
	Controlling Multiple Devices	53
2.6	Concurrent I/O	58
2.7	Summary	65
	Other Reading	65
	Exercises	66
<b>3</b>	<b>PROCESSES</b>	<b>70</b>
3.1	The Process Concept	71
3.2	Systems Programmer's View of Processes	74
	A Multiprocess Example	75
	Interprocess Synchronization	77
	Behavior of Sample Processes	80
	Postlude: Systems Programmer's View of Processes	86
3.3	Operating System's View of Processes	87
3.4	Operating-System Services for Process Management	94
	CREATE	95
	DELETE	95
	FORK/JOIN	96
	ABORT	96
	SUSPEND	97
	RESUME	97
	DELAY	97
	GET_ATTRIBUTES	98
	CHANGE_PRIORITY	98
3.5	Scheduling	99
	Types of Schedulers	100
	The Long-Term Scheduler	100
	The Medium-Term Scheduler	101
	The Short-Term Scheduler	102
	Scheduling and Performance Criteria	103
3.6	Scheduling Algorithms	105
	First-Come-First-Served (FCFS) Scheduling	106
	Shortest Remaining Time Next (SRTN) Scheduling	107
	Time-Slice Scheduling (Round Robin, RR)	111

	Priority-Based Preemptive Scheduling (Event Driven, ED)	116
	Multiple-Level Queue (MLQ) Scheduling	117
3.7	Summary	119
	Other Reading	120
	Exercises	121
<b>4</b>	<b>INTERPROCESS SYNCHRONIZATION</b>	<b>124</b>
4.1	The Interprocess Synchronization Problem	125
	An Example: The Terminal-Handler Problem	125
	Concurrent Updating of a Shared Variable	129
4.2	Mutual Exclusion	132
	The First Algorithm	133
	The Second Algorithm	136
	The Third Algorithm	138
4.3	Semaphores	140
	Semaphore Definition and Busy-Wait Implementation	140
	Some Properties and Characteristics of Semaphores	144
4.4	Hardware Support for Mutual Exclusion	146
	Pessimistic and Optimistic Concurrency Control	146
	Disable/Enable Interrupts	147
	Test-and-Set Instruction	149
	Compare-and-Swap Instruction	151
4.5	Semaphores Revisited	154
4.6	Classical Problems in Concurrent Programming	156
	The Producer/Consumer Problem	157
	Producers and Consumers with an Unbounded Buffer	157
	Producers and Consumers with a Bounded Buffer	161
	Readers and Writers	164
4.7	Summary	167
	Other Reading	167
	Exercises	168
<b>5</b>	<b>INTERPROCESS COMMUNICATION AND SYNCHRONIZATION</b>	<b>175</b>
5.1	Critical Region and Conditional Critical Region	176
5.2	Monitors	178
5.3	Messages	185
	Issues in Message Implementation	186
	Naming	187
	Copying	188
	Buffering	189
	Length	191
	Interprocess Communication and Synchronization with Messages	192
	Interrupt Signaling via Messages	196
5.4	Interprocess Synchronization and Communication in Ada	199
	The Entry-Accept Mechanism	201
	The SELECT Statement	204

5.5	Deadlocks	208
	Reusable and Consumable Resources	210
	Deadlock Prevention	211
	Deadlock Avoidance	213
	Deadlock Detection and Recovery	218
	Combined Approach	221
5.6	Summary	222
	Other Reading	223
	Exercises	224
<b>6</b>	<b>MEMORY MANAGEMENT: CONTIGUOUS ALLOCATION</b>	<b>228</b>
6.1	Single-Process Monitor	231
6.2	Partitioned Memory Allocation–Static	234
	Principles of Operation	234
	Swapping	238
	Relocation	240
	Protection and Sharing	243
	Concluding Remarks	246
6.3	Partitioned Memory Allocation–Dynamic	246
	Principles of Operation	247
	Compaction	252
	Protection and Sharing	254
	Concluding Remarks	257
6.4	Segmentation	258
	Principles of Operation	259
	Protection and Sharing	265
	Concluding Remarks	268
6.5	Summary	269
	Other Reading	270
	Exercises	270
<b>7</b>	<b>MEMORY MANAGEMENT: NONCONTIGUOUS ALLOCATION</b>	<b>273</b>
7.1	Paging	274
	Principles of Operation	274
	Page Allocation	277
	Hardware Support for Paging	278
	Protection and Sharing	280
	Concluding Remarks	282
7.2	Virtual Memory	282
	Principles of Operation	284
	Instruction Interruptibility	285
	Management of Virtual Memory	288
	Program Behavior	289
	Replacement Policies	291
	First-In, First-Out (FIFO)	293

Least-Recently-Used (LRU)	294
Optimal (OPT)	295
Approximations	296
Allocation Policies	297
Page Fault Frequency	300
Working Set Theory	300
Hardware Support and Considerations	302
Protection and Sharing	303
Segmentation and Paging	304
Concluding Remarks	306
7.3 Summary	307
Other Reading	308
Exercises	308
<b>8 FILE MANAGEMENT</b>	<b>312</b>
8.1 Command-Language User's View of the File System	313
8.2 System Programmer's View of the File System	320
8.3 Disk Organization	323
8.4 Disk Controller and Driver	325
8.5 Operating System's View of File-Management	328
Directories	330
Disk-Space Management	335
Contiguous Allocation	336
Noncontiguous Allocation: Chaining	338
Noncontiguous Allocation: Indexing	340
Address Translation	343
File-Related System Services	350
Asynchronous Input/Output	355
8.6 A Generalization of File Services	357
8.7 Summary	358
Other Reading	360
Exercises	360
<b>9 DESIGN OF A KERNEL OF A MULTIPROCESS OPERATING SYSTEM (KMOS)</b>	<b>364</b>
9.1 Defining KMOS Services	366
9.2 Major Design Decisions	368
9.3 Process-State Transitions in KMOS	370
9.4 Functional Specification of KMOS	371
Process Dispatching	372
Interprocess Communication and Synchronization	373
Mailboxes and Messages	373
SEND and RECEIVE Operations	376
Interrupt-Management	378
Process-Management	381
Process Creation	381
Delaying of a Process for a Specified Time	382

	System Startup	383
9.5	Implementation Considerations	386
	Systems-Implementation Languages	387
	Facilities for Modular Program Development	387
	Access to Hardware and to Physical Memory Addresses	389
	Invoking the Operating System	390
	Procedure Call	390
	Supervisor Call	391
	Software Interrupt	392
9.6	Summary	393
	Other Reading	394
	Exercises	394
<b>10</b>	<b>IMPLEMENTATION OF KMOS</b>	<b>397</b>
10.1	KMOS System Lists	398
10.2	Ready List and Its Manipulation	400
	Implementation of the Ready List in KMOS	401
	Process Control-Block	402
	Insertions into the Ready List	404
10.3	Interprocess Communication and Synchronization	407
	Mailboxes and Messages	407
	The SEND-MESSAGE Operation	410
	The RECEIVE-MESSAGE Operation	412
10.4	Interrupt-Management	412
	Interrupt Mailboxes and Priorities	413
	Interrupt Servicing in KMOS	414
	Enabling Hardware Interrupt Levels	417
10.5	Process-Management	417
	Process Dispatching	417
	Static Process-Descriptor	419
	Process Creation	420
	Delaying of a Process for a Specified Time	422
	The Delayed List	422
	The DELAY Operation	427
	Timer-Interrupt Processing	428
10.6	Startup and Initial System Configuration	429
10.7	Summary	431
	Other Reading	432
	Exercises	432
	KMOS Code	436
<b>11</b>	<b>CASE STUDIES</b>	<b>449</b>
11.1	PC-DOS (MS-DOS) Operating System	449
	Command-Language User's View of the PC-DOS	450
	System-Call User's View of the PC-DOS	452
	PC-DOS Implementation	453
	PC-DOS Summary	456

11.2	Unix Operating System	457
	Command-Language User's View of Unix	458
	System-Call User's View of Unix	463
	Unix Implementation	467
	Unix Summary	471
11.3	iRMX 86 Operating System	472
	Command-Language User's View of the iRMX 86	474
	System-Call User's View of the iRMX 86	475
	iRMX 86 Implementation	481
	iRMX 86 Summary	482
11.4	Design of A Remote-Telemetry Unit (RTU)	484
	Computer Data-Acquisition and Control Systems	484
	The Role of a Remote-Telemetry Unit	487
	Functional Organization and Activities of an RTU	490
	RTU Software Organization (Processes)	493
	RTU Data Structures and Interprocess Communication	496
	Operation of RTU Processes	498
	RTU Processes and KMOS	502
	Concluding Remarks	503
	Other Reading	505
	<b>Appendix A: Assembly-Language Portion of KMOS Code</b>	506
	<b>Appendix B: Configuration Macros</b>	527
	<b>Bibliography</b>	540
	<b>Index</b>	551