## Inhaltsverzeichnis

Vorwort			9
Kapitel	1	Einleitung	11
Kapitel	2	Grundlagen	15
	2.1	Betriebssysteme	15
	2.1.1	Multitasking-Multiuser-Systeme	17
	2.1.2	Timesharing-Realtime-Systeme	19
Kapitel	3	iRMX-86 als Realtime-Multitasking-System	21
	3.1	Aufbau	23
	3.2	Hardware-Komponenten	23
	3.3	Aufgaben der Systemkomponenten	24
	3.4	Schnittstellen zu den Applikationen	26
	3.5	Der Nucleus	28
	3.5.1	Objekte	28
	3.5.2	Tasks (Prozesse)	29
		Multitasking	31
		Scheduling	32
	3.5.3	Kommunikation	34
		Allgemeines	34
		Segmente	34
		Mailboxen	35
	3.5.4	Synchronisation	37
		Allgemeines	37
		Semaphore und Regions	38
		Globale Semaphore	39
		Systemverklemmungen	40
	3.5.5	Verwaltungsstrukturen	42
		Allgemeines	42
		Jobs	43

	3.5.6	Systemerweiterungen	45
	3.5.7	Fehlerbehandlung	46
		Zentrale Fehlerbehandlung	47
		Taskinterne Fehlerbehandlung	48
	3.5.8	Interruptverarbeitung	48
		Die Interrupt-Hardware	48
		Interrupt-Service-Routine und Interrupt-Task	54
	3.5.9	iRMX-86 auf Multibus II	59
		Allgemeines	59
		Kommunikation im Multicomputerbereich	60
	3.5.10	Testhilfen	62
		Der System-Debugger	62
		Der Dynamic-Debugger	63
	3.6	Der Terminal-Handler	64
	3.7	Die Eingabe-Ausgabe-Systeme	65
	3.7.1	Allgemeines	65
	3.7.2	Das Basic-I/O-System	66
		Files	68
		User	70
		I/O-Operationen	71
		Filetreiber	72
		Gerätetreiber	74
		Schnittstellen zu den Gerätetreibern	76
		Der Custom-Treiber	77
		Die Common- oder Random-Treiber	79
		Das Extended-I/O-System	80
		Logische Namen	81
		Puffer im Extended-I/O-System	82
		I/O-Jobs	83
	3.8	Der Applicationloader	84
	3.8.1	Allgemeines	84
	3.8.2	Laden von Code	86
Kapitel	4	iRMX-86 als Multiuser-Multitasking-	
-		Entwicklungssystem	87
	4.1	Das Human-Interface	87
	4.1.1	Die Kommandozeilen-Interpreter (CLI)	81
	4.1.2	Das Filesystem	89
	4.1.3	Standardkommandos	90
		Kommandos für File-Management	9
		Kommandos für Volume-Management	9:
		Kommandos für Multiuser-Betrieb	99
		Kommandos des Systemmanagers	100
		Kommandos für die generelle Systemsteuerung	10

	4.1.4	Kommandos des Kommandointerpreters (CLI) UDI (Universal-Development-Interface)	106 108
Kapitel	5	Erstellung von Multitasking-Realzeit-	
		Applikationen	111
	5.1	Strukturiertes Design	111
	5.2	Prioritäten	114
	5.3	Leistungsverbesserungen	114
Kapitel	6	Systemkonfiguration	117
	6.1	Hardware	118
	6.2	ICU (Interactive-Configuration-Utility)	119
	6.2.1	iRMX-Definitionsfiles	121
	6.2.2	Memorydesign	125
	6.2.3	Applikationen als First-Level-User-Jobs	127
Kapitel	7	Der Bootstraploader	131
Kapitel	8	Systemcalls an iRMX aus der Programmier-	
		sprache »C«	133
	8.1	Allgemeines	133
	8.2	Entwicklungs-Software	134
	8.3	Abbildung der »C«-Datentypen auf die RMX-Datentypen	135
	8.4	iRMX-Strings in »C«	135
	8.5	Parameterübergabe an die Systemcalls	135
	8.6	Zugriff auf dynamisch bereitgestelltes Memory	136
	8.7	Kreieren von iRMX-Objekten und Operationen auf diesen	
		Objekten	137
	8.8	Entwicklung einer Applikations-Software in »C« unter iRMX	
		als Entwicklungssystem	138
	8.9	Testen einer Applikation unter dem Human-Interface	139
	8.9.1	Testen mit dem SDB unter dem DEBUG-Kommando	139
	8.9.2	Testen der Applikations-Software mit PSCOPE	140
	8.10	Entwicklung einer Applikations-Software in »C« unter iRMX-86	
		und Einkonfiguration als First-Level-User-Job	141
Kapitel	9	Von iRMX-86 zu iRMX-286	145
	9.1	Was ist iRMX-286?	145
	9.2	Portierung der Applikations-Software von iRMX-86	140
	0.2	zu iRMX-286	148
	9.3	Nutzung neuer iRMX-286-Features	152
	9.4	Systemkonfiguration	153
	9.5	iRMX-286-Realtime-Scheduling <—> Timesharing für Multiuser-Umgebung	155
		vuutuser-t/mgening	177

Kapitel	10	Systeminstallation und Systemwartung	157
-	10.1	Systeminstallation	157
	10.2	Systemwartung (Updates)	158
Anhang	Α	Interface-Bibliotheken zu den iRMX-86-Subsystemen	161
	В	Kurzbeschreibung aller iRMX-Systemcalls	163
	C	Die Systemcalls aus der Programmiersprache PL/M	173
	D	Die Systemcalls aus der Programmiersprache Pascal	227
	Е	Die Files mit den External-Deklarationen der Systemcalls für	
		die Programmiersprachen PL/M-86, Pascal, Fortran, iRMX-286	257
	F	Stacksegmentbedarf der Systemcalls	265
	G	Speicherbedarf der iRMX-Objekte	269
	Н	Maximaler Speicherbedarf der iRMX-Subsysteme	271
	I	Programmierbeispiele in PL/M mit den iRMX-Definitionsfiles	273
		1. Intertaskkommunikation	273
		2. Tasksynchronisation	279
		3. Ein-/Ausgabe mit dem Basic-I/O-System	286
		4. Konfiguration von eigenen Gerätetreibern	298
		5. Ein-/Ausgabe mit dem Extended-I/O-System	305
		6. Definitionsfile für ein iRMX-86-Human-Interface-System für	
		ein Intel-Computersystem 286/310	313
		7. Interruptverarbeitung unter iRMX-286 mit Interrupt-Service-	
		Routine und Interrupt-Task	348
Stichwo	rtverzo	eichnis	359
Hinweise	e auf v	weitere Markt&Technik-Produkte	362