

Inhaltsverzeichnis

1	Einleitung	1
1.1	Programmierparadigmen	2
1.2	Parallelitätsebenen	5
1.3	Homogene bzw. heterogene Struktur des Mehrrechnersystems	7
1.4	Verbindungs-Netzwerke	8
1.5	Garbage Collection	13
1.6	Overhead	15
1.7	Literaturverzeichnis	16
2	Prozedurale Programmiersprachen (Ada, CHILL)	18
2.1	Einführung und Sprachmittel der höheren Programmiersprachen Ada und CHILL	20
2.1.1	Ada	20
2.1.1.1	Einführung in Ada	20
2.1.1.2	Ada – Allgemeine Sprachmittel	22
2.1.1.3	Ada – Sprachmittel für Parallelität	26
2.1.2	CHILL	35
2.1.2.1	Einführung in CHILL	35
2.1.2.2	CHILL – Allgemeine Sprachmittel	36
2.1.2.3	CHILL – Sprachmittel für Parallelität	41
2.1.2.4	Kritikpunkte	45
2.2	Parallelitätseigenschaften	45
2.2.1	Ada	46
2.2.1.1	Ada – Expliziter Parallelismus	46
2.2.1.2	Ada – Impliziter Parallelismus	46
2.2.1.3	Ada – Granularität	48
2.2.2	CHILL	49
2.2.2.1	CHILL – Expliziter Parallelismus	49
2.2.2.2	CHILL – Impliziter Parallelismus	49
2.2.2.3	CHILL – Granularität	50
2.3	Kommunikation und Synchronisation	51
2.3.1	Ada	51
2.3.1.1	Ada – Kommunikation	51
2.3.1.2	Ada – Synchronisation	52
2.3.2	CHILL	53
2.3.2.1	CHILL – Kommunikation	53
2.3.2.2	CHILL – Synchronisation	55
2.3.2.2.1	Explizite Synchronisation	56
2.3.2.2.2	Implizite Synchronisation	57
2.4	Parallelitätsbedingte Verwaltungsaufgaben	57
2.4.1	Prozeßverwaltung in Ada	57
2.4.2	Prozeßverwaltung in CHILL	62
2.4.3	Speicherverwaltung in Ada und CHILL	65

2.5	Abstrakte Maschine	69
2.5.1	Die Zwischensprachen für Ada	71
2.5.1.1	DIANA	71
2.5.1.2	LOLITA	71
2.5.1.3	I-Code und die abstrakte Maschine A0	72
2.5.2	Die Zwischensprache als Befehlssatz der CHILL-Stackmaschine	75
2.5.2.1	Speicherverwaltung	75
2.5.2.2	Programmauswertung	75
2.5.2.3	Datentypen	77
2.5.2.4	Befehlssatz	77
2.5.2.5	Bewertung der Zwischensprache	78
2.6	Hardware-Architekturen	80
2.6.1	Intellimac IN/7000 (Ada)	80
2.6.2	Rational R 1000 (Ada)	82
2.6.3	Intel iAPX 432	84
2.6.4	K-CHILL-Rechner	91
2.6.5	Integrierter CHILL-Prozessor	91
2.7	Resümee	93
2.8	Literaturverzeichnis	96
3	Objektorientierte Programmiersprachen	100
3.1	Einführung und Sprachmittel	100
3.1.1	Konzepte der objektorientierten Sprachen am Beispiel Smalltalk	100
3.1.1.1	Objekte und Nachrichten – Klassen und Instanzen	101
3.1.1.2	Dynamisches Binden und Vererbung	103
3.1.1.3	Objektorientierte Ansätze in anderen Sprachen ..	105
3.1.2	Sprachmittel für Parallelität	105
3.1.2.1	Explizite Prozesse in Smalltalk-80	105
3.1.2.2	Implizite Prozesse auf Objektebene	110
3.1.2.2.1	Synchrone Botschaften	110
3.1.2.2.2	Asynchrone Botschaften	111
3.2	Parallelitätseigenschaften	116
3.2.1	Impliziter Parallelismus	116
3.2.2	Explizite Parallelität	117
3.2.3	Instantiierung und Terminierung	118
3.2.4	Granularität	118
3.3	Kommunikation und Synchronisation	118
3.3.1	Kommunikation	119
3.3.1.1	Botschaften in Smalltalk	119
3.3.1.2	Prozeßkommunikation in Smalltalk	120
3.3.1.3	Asynchrone Kommunikation bei Objekten	120
3.3.1.4	Kommunikation in Aktor-Systemen	122
3.3.2	Synchronisation	124
3.3.2.1	Semaphore in Smalltalk	124
3.3.2.2	Synchronisation bei asynchronen Botschaften ...	125
3.4	Parallelitätsbedingte Verwaltungsaufgaben	125
3.4.1	Prozeßverwaltung	125
3.4.1.1	Prozeßverwaltung in Smalltalk-80	125
3.4.1.2	Prozesse auf Objektebene	127

3.4.2	Speicherverwaltung	128
3.5	Abstrakte Maschine	130
3.5.1	Die virtuelle Maschine von Smalltalk	130
3.5.2	Abstrakte parallele Maschinen	133
3.5.3	Zwischensprache	135
3.6	Hardware-Architekturen	137
3.6.1	Smalltalk mit RISC	137
3.6.2	Intel iAPX 432	140
3.6.3	Die FAIM-1 Architektur	144
3.7	Resümee	146
3.8	Literaturverzeichnis	147
4	Funktionsbasierte Programmiersprachen	150
4.1	Einführung und Sprachmittel	150
4.1.1	Funktionale Programmierung	150
4.1.2	Die Programmiersprache Lisp	152
4.1.2.1	Dialekte und Standardisierung	153
4.1.2.2	Prinzipien der Programmiersprache Lisp	153
4.1.2.3	Anwendungsgebiete von Lisp	161
4.1.3	Sprachmittel für Parallelität	161
4.2	Parallelitätseigenschaften	169
4.2.1	Impliziter Parallelismus	171
4.2.1.1	Parallele Termevaluierung	172
4.2.1.2	Wertunabhängige Operationen	176
4.2.1.3	Programmierstile	177
4.2.1.4	Implementierung	178
4.2.2	Expliziter Parallelismus	178
4.2.3	Granularität	179
4.2.4	Weitere Ansätze zur Steigerung der Parallelität ..	180
4.3	Kommunikation und Synchronisation	182
4.3.1	Kommunikation	182
4.3.2	Synchronisation	183
4.3.3	Einfluß der Granularität	183
4.4	Parallelitätsbedingte Verwaltungsaufgaben	184
4.4.1	Prozeßverwaltung	184
4.4.2	Speicherverwaltung	190
4.5	Abstrakte Maschine	191
4.5.1	Struktur der abstrakten Maschine	192
4.5.2	Befehle der abstrakten Maschine	199
4.6	Hardware-Architekturen	201
4.6.1	Architektur-Vorschläge	201
4.6.1.1	Ein Multiprozessor-System für prozeß-orientiertes Lisp	201
4.6.1.2	Eine listenverarbeitende Datenfluß-Maschine ...	205
4.6.2	Leistungsfähigkeit funktionaler Parallelrechner ..	208
4.7	Resümee	210
4.8	Literaturverzeichnis	212
5	Logische Programmiersprachen	214
5.1	Einführung in Prolog	214
5.1.1	Sprachkonstrukte und Semantik	214

5.1.1.1	Definitionen	214
5.1.1.2	Syntaktische Konstrukte	215
5.1.1.3	Semantische Besonderheiten von Prolog	218
5.1.2	Impliziter Parallelismus in Prolog	221
5.1.3	Untersuchungen über den Aufbau von Prolog- programmen	223
5.2	Vom ODER-Parallelismus beeinflusste Ansätze zur parallelen Abarbeitung von Prolog	225
5.2.1	Sprachmittel	225
5.2.1.1	Sprachmittel für Parallelität	227
5.2.1.2	Statische Befehlshäufigkeit	230
5.2.2	Parallelitätseigenschaften	232
5.2.2.1	Impliziter Parallelismus	232
5.2.2.2	Expliziter Parallelismus	235
5.2.2.3	Granularität	235
5.2.3	Kommunikation und Synchronisation	236
5.2.3.1	Kommunikation	236
5.2.3.2	Synchronisation	237
5.2.4	Parallelitätsbedingte Verwaltungsaufgaben	238
5.2.4.1	Prozeßverwaltung	238
5.2.4.2	Speicherverwaltung	239
5.2.4.3	Verschiedene Modelle der ODER-parallelen Verarbeitung	240
5.2.4.3.1	Kabu Wake	240
5.2.4.3.2	PEPSys	242
5.2.4.3.3	OR-Parallel Token Machine	249
5.2.4.3.4	ODER-parallele Ausführung auf einer multi- sequentiellen Maschine	251
5.2.5	Abstrakte Maschine	254
5.2.5.1	Die sequentielle Warren-Maschine	255
5.2.5.2	Zwischensprache	262
5.2.6	Hardware-Architekturen	264
5.2.6.1	Architekturvorschläge	265
5.2.6.2	Beurteilung	270
5.2.7	Resümee	271
5.3	Committed-Choice Sprachen	273
5.3.1	Sprachmittel	275
5.3.1.1	Überblick	275
5.3.1.2	Parallelkonstrukte	276
5.3.1.2.1	Guarded Horn Clauses	276
5.3.1.2.2	Sprachkonstrukte für die Synchronisation	278
5.3.1.2.3	Die Sprache KP <small>UND</small> -Baum	283
5.3.1.2.4	Die Verwendung der Variablen als Kommunikationsmittel	285
5.3.2	Parallelitätseigenschaften	287
5.3.2.1	Impliziter Parallelismus	287
5.3.2.2	Explizite Parallelität	290
5.3.2.3	Granularität	290
5.3.3	Kommunikation und Synchronisation	291
5.3.3.1	Kommunikation	291
5.3.3.2	Synchronisation	293

5.3.4	Parallelitätsbedingte Verwaltungsaufgaben	294
5.3.4.1	Prozeßverwaltung	295
5.3.4.2	Speicherverwaltung	301
5.3.5	Abstrakte Maschine	303
5.3.5.1	Die abstrakte Maschine von Levy für GHC	303
5.3.5.2	Die abstrakte Maschine für das Datenflußmodell	314
5.3.5.3	Die abstrakte Maschine zur KP _{UND} -Baum- Sprache	318
5.3.6	Hardware-Architekturen	319
5.3.6.1	PIM-R	319
5.3.6.2	PIM-D	327
5.3.6.3	Die neue PIM-Maschine von ICOT	331
5.3.6.4	ALICE	332
5.3.7	Resümee	333
5.4	PROLOG-Anwendungen und Datenbanken	334
5.4.1	Motivation	334
5.4.2	Problemstellung	335
5.4.2.1	Darstellung von Fakten und Regeln	335
5.4.2.2	Schnittstellen zwischen DBMS und KBS	338
5.4.2.3	Ausblick	340
5.5	Literaturverzeichnis	341
6	Vergleich der Architekturvorschläge	346
7	Glossar	350
	Sachverzeichnis	357