

Axel Groha

**Universelles
Zellenrechnerkonzept für
flexible Fertigungssysteme**

Mit 74 Abbildungen

**Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo 1988**

Inhaltsverzeichnis

	Seite
1. Einleitung und Zielsetzung	1
2. Anforderungen an die Informationsstruktur flexibler Fertigungssysteme	3
2.1. Problemstellung	3
2.2. Entkopplung der Systemkomponenten durch eine sinnvolle Informationsstruktur	4
2.3. Topologie für den Informationsfluß im Rechner- und Steuerungsverbund eines flexiblen Fertigungssystems	6
3. Der Zelligedanke als Strukturierungsprinzip im Informations- und Materialfluß flexibler Fertigungssysteme	13
3.1. Einführung der flexiblen Zelle	13
3.2. Zellenrechnerebene im flexiblen Fertigungssystem	13
3.3. Trend zur rechnergeführten Einmaschinenzelle	14
3.4. Struktur einer flexiblen Bearbeitungszelle	16
3.5. Analogiebetrachtungen für die flexible Montagezelle	18
3.6. Allgemeine flexible Fertigungszelle	22
3.7. Horizontale Informationsstruktur der Zellenrechnerebene	25
3.8. Die Zellenrechnerebene als Hierarchieebene des vertikalen Informationsflusses	26
4. Aufgaben einer eigenständigen Materialflußzelle im Informations- und Materialfluß flexibler Fertigungssysteme	28
4.1. Einführung der flexiblen Materialflußzelle zur strukturellen Komplettierung	28
4.2. Struktur einer flexiblen Materialflußzelle	29
4.3. Aufgabenverteilung zwischen Materialflußzellenrechner und Leitreechner im horizontalen und vertikalen Informationsfluß	30
4.4. Funktionen des horizontalen Informationsflusses in der Zellenrechnerebene	32

	Seite
5. Aufgabenspektrum eines universellen Fertigungszellenrechners	37
5.1. Übersicht	37
5.2. Kernaufgaben eines universellen Fertigungszellenrechners	38
5.2.1. Auftragseinplanung	39
5.2.2. Auftragsdurchführung	41
5.2.3. Diagnose	42
5.3. Dienstleistungsmodulare für die Kernaufgaben	44
6. Der Zellenrechner-Systemprogrammkernel	46
6.1. Möglichkeiten zur Optimierung der Auftragsabwicklung	46
6.1.1. Optimierung nach minimalem Rüstaufwand	46
6.1.2. Optimierung aufgrund der funktionalen Gliederung der Auftragsdurchführung	47
6.1.3. Unterstützung von Nebenläufigkeiten in der Auftragsdurchführung	49
6.2. Geeignete Prozeßarchitektur für die Realisierung paralleler Aufgaben	51
6.3. Die Auftragsabwicklung	54
6.3.1. Einlastungsphase	54
6.3.2. Dispositionsphase	55
6.3.3. Vorbereitungsphase	57
6.3.4. Bearbeitungsphase	61
6.3.5. Einfluß der Auftragsprioritäten auf die gesamte Auftragsabwicklung	62
6.4. Prozeßarchitektur des Zellenrechner-Systemprogrammkernelns	67
6.5. Kommunikation mit dem Leitrechner	69
6.6. Kommunikation mit dem Materialflußzellenrechner	73
6.7. Dialog mit dem Bediener	74
6.8. Statistische Datenauswertung und Datenkonzentration	78
7. Realisierung der Ablaufsteuerung	79
7.1. Definitionen	79
7.2. Offenes Realisierungskonzept	81
7.3. Schnittstelle zwischen Zellenrechner-Systemprogramm und Komponententreibern	82

	Seite	
7.4.	Universelles Konzept für die funktionsorientierte Ablaufsteuerung	83
7.4.1.	Funktionsorientiertes Prinzip für den Entwurf der Ablaufsteuerung	83
7.4.2.	Parametrisierbarkeit der fest eingepprägten Funktionssteuerung	84
7.4.3.	Öffnung der Funktionssteuerung für die Anwenderprogrammierung	85
7.5.	Universelles Konzept für die aktionsorientierte Ablaufsteuerung	86
7.6.	Zustandsorientierte Aktionssteuerung	89
7.7.	Koordination rechnerexterner Parallelitäten in der Aktionssteuerung	93
7.8.	Petri-Netz-orientierte Aktionssteuerung	94
7.8.1.	Überlegenheit der Petri-Netze für den Steuerungsentwurf	94
7.8.2.	Theoretische Grundlagen der Petri-Netze	95
7.8.3.	Anwendbarkeit der Petri-Netze auf die Aktionssteuerung	96
7.8.4.	Programmwurf der Ablaufvorschrift	99
7.8.5.	Textuelle Darstellung der Petri-Netz-orientierten Ablaufvorschrift	104
7.8.6.	Programminterne Darstellung der Petri-Netz-orientierten Ablaufvorschrift	107
7.8.7.	Algorithmus der Aktionssteuerung	111
7.9.	Wirkungsweise der Ereignispufferungsebene	113
7.10.	Zeitüberwachung für jede gestartete Aktion	115
8.	Realisierung der erweiterten Komponententreiber	117
8.1.	Begriffserklärung	117
8.2.	Realisierung der Standardschnittstelle zwischen Zellenrechner- Systemprogrammkerneln und den erweiterten Komponententreibern	117
8.2.1.	Transparente Standardschnittstelle	117
8.2.2.	Aktionsaufruf und Aktions-Fertigmeldung	119
8.2.3.	Genormte Aufbereitung der vom technischen Prozeß benötigten Informationen	120
8.3.	Ausführung rechnerinterner Aktionen in der Rechnerkomponente	121
8.4.	Aufruf der Funktionen zum Bedarfsabgleich im Anwenderprogramm	124
8.5.	Erweiterte Komponententreiber als eigenständige Programme	125
8.6.	Aufbau eines erweiterten Komponententreibers	126

	Seite
9. Generierung und Programmierung des Zellenrechner-Systemprogramms durch den Anwender	130
9.1. Verlagerung aller Konfigurations- und Auftragsabhängigkeiten vom Systemprogrammkernel in externe Strukturen	130
9.2. Generierung des Zellenrechner-Systemprogramms durch den Anwender	132
9.3. Programmierung der Ablaufvorschrift durch den Anwender	133
9.3.1. Personelle Zuordnung, Art und Ort der Anwendungsprogrammierung	133
9.3.2. Problemorientierte Zellenprogrammierung	135
10. Zusammenfassung und Ausblick	138
11. Literaturverzeichnis	140