

Inhalt

| | |
|---|-----|
| Vorwort | 1 |
| 1. Grundlagen | 9 |
| 1.1 Steuerungsaufgaben lösen | 9 |
| 1.2 Ein Blick in die Entwicklungsgeschichte | 15 |
| 1.3 Speicher- oder Schaltungsprogrammierung? | 18 |
| 2. Zustandsautomaten und Steuerautomaten | 25 |
| 2.1 Der Steuerautomat als Zustandsautomat | 25 |
| 2.1.1 Einführung | 25 |
| 2.1.2 Zustände und Zustandsübergänge | 28 |
| 2.1.3 Einfache Zustandsübergänge | 30 |
| 2.1.4 Zeitplansteuerungen (Sequencers) | 33 |
| 2.1.5 Universelle Steuerautomaten | 38 |
| 2.1.6 Zustandsautomaten aus der Verhaltensbeschreibung synthetisieren | 45 |
| 2.2 Grundsatzlösungen der Speicherprogrammierung | 59 |
| 2.2.1 Speicherbasierte Zustandsautomaten | 60 |
| 2.2.2 Algorithmische Zustandsautomaten | 65 |
| 2.3 Boolesche Steuerautomaten | 71 |
| 2.3.1 Einführung | 71 |
| 2.3.2 Mit Bits rechnen | 75 |
| 2.3.3 Wertetabellen auslesen | 81 |
| 2.4 Universelle algorithmische Automaten | 86 |
| 2.5 Die Turing-Vollständigkeit | 95 |
| 3. Universalrechner als Steuerautomaten | 107 |
| 3.1 Die Universalmaschine oder etwas anderes? | 107 |
| 3.2 Universalprozessoren einsetzen | 110 |
| 3.2.1 EDV-Maschinen und programmierbare Emulatoren | 110 |
| 3.2.2 EDV-Maschinen als Steuerautomaten | 113 |
| 3.2.3 Steuerungsaufgaben erledigen | 121 |
| 3.3 Prozessoren und programmierbare Logik | 125 |
| 3.4 Hardware und Software | 129 |
| 3.5 Universal- und Spezialmaschinen | 132 |
| 3.6 Der Universalprozessor als Schaltungsbaustein | 146 |
| 3.7 Multitasking in der Hardware | 152 |
| 4. Grundlagen der Mikroprogrammsteuerung | 159 |
| 4.1 Zur Entwicklungsgeschichte | 159 |
| 4.1.1 Das Steuerwerk als Entwurfsaufgabe | 159 |
| 4.1.2 Direkte Steuerung | 161 |
| 4.1.3 Sequentielle Steuerung | 163 |

| | |
|---|------------|
| 4.1.4 Steuerketten | 163 |
| 4.1.5 Mikroprogrammsteuerung | 166 |
| 4.2 Der Computer im Computer | 169 |
| 4.3 Mikrobefehle im Operationsautomaten | 181 |
| 4.4 Leistungsprobleme der Mikroprogrammsteuerung | 186 |
| 4.4.1 Verzweigungen | 186 |
| 4.4.2 Mit Parametern arbeiten | 188 |
| 4.4.3 Wandeln und adaptieren | 190 |
| 4.5 Mikrobefehlsformate | 192 |
| 4.6 Nanoprogrammsteuerung und Nanobefehle | 198 |
| 4.7 Maschinenbefehle und Mikrobefehle | 201 |
| 4.8 Mikrobefehle als Grundlage der Maschinenarchitektur | 206 |
| 5. Mikroprogrammsteuerwerke | 213 |
| 5.1 Mikroprogrammsteuerwerke entwerfen | 213 |
| 5.1.1 Die Ressourcen entscheiden alles | 213 |
| 5.1.2 Der Mikrobefehlszyklus | 215 |
| 5.1.3 Bedingungssignale | 216 |
| 5.1.4 Das Taktsystem | 218 |
| 5.2 Mikrobefehlsadressierung | 228 |
| 5.2.1 Nachfolgende Mikrobefehle auswählen | 228 |
| 5.2.2 Segmentierung | 237 |
| 5.2.3 Adreßzählung | 244 |
| 5.2.4 Folgeadressen im Mikrobefehl | 262 |
| 5.2.5 Funktions- und Mehrwegeverzweigung | 269 |
| 5.2.6 Spätverzweigung | 278 |
| 5.2.7 Segmentierung | 280 |
| 5.3 Wartezustände und Wartemikrobefehle | 284 |
| 5.4 Mikroprogrammunterbrechungen | 289 |
| 5.5 Unterprogramme | 312 |
| 6. Maschinen mit Mikroprogrammsteuerung | 319 |
| 6.1 Zeitplansteuerungen (Sequencers) | 320 |
| 6.2 Universelle Steuerautomaten (Branch Sequencers) | 332 |
| 6.3 Algorithmische Steuerautomaten | 343 |
| 6.4 Universalmaschinen | 360 |
| 6.4.1 Einadreßmaschinen | 361 |
| 6.4.2 Universalregistermaschinen | 377 |
| 6.4.3 Mikrobefehle als Maschinenbefehle | 386 |
| 7. Der Prozessor als Mikroprogrammsteuerwerk | 401 |
| 7.1 Eine alternative Design-Idee | 401 |
| 7.2 Nebenläufige Wirkungen | 408 |

| | |
|---|------------|
| 7.2.1 Unterbrechungen zeitweilig verhindern | 409 |
| 7.2.2 Register nebenher laden | 410 |
| 7.2.3 Zusatzausgabe | 411 |
| 7.3 Nebenläufige Ausgabe | 412 |
| 7.3.1 Datenausgabe | 413 |
| 7.3.2 Direktwertausgabe | 413 |
| 7.3.3 Adreßausgabe | 415 |
| 7.4 Dateneinspeisung | 418 |
| 7.4.1 Eingabe in die Prozessorregister | 419 |
| 7.4.2 Eingabe in den Arbeitsspeicher | 419 |
| 7.4.3 Adressen einspeisen | 421 |
| 7.5 Befehlsmodifikation | 421 |
| 7.5.1 Befehlssubstitution | 423 |
| 7.5.2 Bedingte Befehlsausführung | 424 |
| 7.5.3 Funktionsverzweigung | 425 |
| 7.5.4 Der EXECUTE-Befehl | 426 |
| 7.5.5 Befehlsausgabe | 426 |
| 7.6 Anwendungsbeispiele | 427 |
| 7.6.1 Akzeleratoren anschließen | 427 |
| 7.6.2 Zustandsautomaten unterstützen | 430 |
| 7.6.3 Den Prozessor auf Kommandosteuerung umbauen | 432 |
| 7.6.4 In den Prozessorkern eingreifen | 432 |
| 7.7 Der Erweiterungs- oder Steuerspeicher | 434 |
| 8. Schaltungseinzelheiten | 439 |
| 8.1 Steuerspeicher | 439 |
| 8.2 Diagnose und Debugging | 453 |
| 8.3 Schnell und schmal oder langsamer und breit? | 467 |
| 8.4 Zeitintervalle darstellen | 470 |
| 8.5 Synchronisieren | 473 |
| 8.6 Taktphasen | 479 |
| 8.7 Der Universalprozessor als Schaltungsbaustein | 483 |
| 9. Historische Beispiele | 495 |
| 9.1 Mainframes | 495 |
| 9.1.1 IBM System /360 | 495 |
| 9.1.2 IBM System /370 | 506 |
| 9.1.3 RCA Spectra 70/45 | 512 |
| 9.2 Kleinere Maschinen | 514 |
| 9.2.1 CDC 5600 | 514 |
| 9.2.2 Honeywell LSI-6 | 520 |

| | |
|---|------------|
| 9.3 Sequencer | 524 |
| 9.3.1 Altera EPS448 SAM EPLD | 524 |
| 9.3.2 AMD Am29C331 | 529 |
| Anhang 1 Eigene Maschinen programmieren | 537 |
| Anhang 2 Elementare Operationen | 539 |
| 2.1 Universelle Operationswerke | 540 |
| 2.2 Binärzahlen addieren und subtrahieren | 542 |
| 2.3 Das herkömmliche Registermodell der Multiplikation und Division | 544 |
| 2.4 Verschieben und Rotieren | 548 |
| Literatur- und Quellenverzeichnis | 551 |
| Index | 561 |