M. BÖHNKE
Siemens AG Karlsruhe

Grundsatzsoftware
320/330

Grundsatz-Software 320/330
Kurzfassung des Vortrags vom 11.4.1973

Assembler- und Makrosprache

1. Eigenschaften des Assemblers

- a) Komfort in der Programmerstellung:
Durch den Matrixaufbau der Befehlsliste und die Notation mit Operatoren kann die Sprache schnell erlernt und bequem hantiert werden.
- b) Problem-Transparenz:
Vergleichbar mit der Notation in höheren Sprachen.
- c) Grundsprache-Transparenz:
Die für die Assemblersprache typische Grundsprache-Transparenz ist beim Assembler 320/330 erhalten worden. Es können sämtliche Maschinenbefehle eingesetzt und die Möglichkeiten der Speicherplatz- und Ablaufzeit-Optimierung genutzt werden.

2. Eigenschaften der Makrosprache

- a) Möglichkeiten für den Einsatz:
 - Betriebssystemaufrufe
 - Erweitern des Assembler-Sprachumfangs
 - Textgenerator
 - Generator für Anwenderprogrammsysteme

b) Komfort:

Hoher Komfort durch die Möglichkeiten der aktiven Hantierung während des Übersetzungslaufes.

Folgende Funktionen sind Bestandteil der Sprache:

Vergleichen von Zeichenfolgen;
bedingtes Absetzen von Zeichenfolgen;
wiederholtes Absetzen von Programmteilen oder Zeichenfolgen;
Generieren von Zeichenfolgen;
Rechnen mit Zeichenfolgenlängen und Parameterzahlen.

c) Unabhängigkeit von Maschinenstrukturen:

Als Objektaufgabe des Makroübersetzers sind Maschinencode für beliebige Maschinenstrukturen, Notationen in beliebigen Assembler- oder Compilersprachen, Texte, Tabellen, Funktionsblöcke o. ä. möglich.

3. Zukunftssicherheit von Assembler- und Makrosprache:

Neue Rechnerentwicklungen bedingten bisher eine neu zu entwickelnde Grundsatz- und Anwendersoftware. Sie binden damit bei Rechnerherstellern und -Anwendern große Entwicklungskapazitäten und finanzielle Mittel.

Durch die unter 1. b und 2. c beschriebenen Eigenschaften und die damit erreichte weitgehende Unabhängigkeit von Maschinenstrukturen erzielen in Assembler- und Makrosprache 320/330 geschriebene Programme eine hohe Zukunftssicherheit, die mit in Compilersprachen notierten Programmen vergleichbar ist.

Organisationsprogramm 320/330

1. Aufbau Funktionsmodular, ORG-Generierung durch Anwender

Für den Anwender steht mit der Anlage generell ein Masterstapel mit allen verfügbaren Funktionen bereit. Entsprechend der gewünschten Komfortstufe und der eingesetzten Hardware-Konfiguration sind hieraus jederzeit neue Organisationsprogramme generierbar.

2. Ablaufobjekte

a) Programm:

Kleinste, selbständig ablauffähige Einheit;
Verwaltung der Simultanarbeit durch das ORG.

b) Aufgabe:

Sequenzieller Ablauf von Funktionskomplexen im Rahmen eines Programms.

c) Common-Code:

Ablauf-invarianter Code.

Kann von beliebig vielen Programmen simultan benutzt werden.

d) Simulationsroutine:

Wird durch einen von der Hardware nicht interpretierbaren Befehl aktiviert.

Ablauf-invarianter Code.

Kann von beliebig vielen Programmen simultan benutzt werden.

3. Beliebig viele Programmnummern / 255 Prioritäten

Die Zuordnung von Programmnummern und Prioritäten ist frei wählbar.

4. Hauptspeicher-Belegung

Die Belegung des Hauptspeichers durch Programme und Dateien kann zentral durch das ORG durchgeführt werden.

5. Datenübergabe

Das ORG steuert die Datenübergabe zwischen Programmen.

6. Funktionen für Programmkoordinierung

Das Organisationsprogramm 320/330 zeichnet sich durch einen ungewohnt hohen Koordinierungskomfort aus.

a) Beispiel "Koordinierungszähler":

Steuerung des Zugriffs mehrerer Anwender zu einem Objekt. Objekte: Geräte, Dateien, Programme.

7. Ein-Ausgabe-Aufrufe

a) Standardaufrufe:

Programme können geräteunabhängig erstellt werden.

Der Anwender braucht die physikalischen Eigenschaften der Geräte nicht zu berücksichtigen.

b) Sonderaufrufe:

Spezielle physikalische Eigenschaften der Sonderperipherie werden angesprochen.

8. Real-Zeit-Organisation

Verzögerter Aufruf von Programmen.

Zyklischer Aufruf von Programmen.

Aufruf von Programmen zu absolut vorgegebener Zeit.

Überwachung des Programmlaufs.

9. Wiederanlauf

Der Wiederanlauf nach Netzausfall erfolgt hardware-unterstützt durch das ORG. Für anwenderspezifische Anlauffunktionen sind Nahtstellen definiert.

10. Dialog

Realisierung des Dialogverkehrs von Bediengeräten mit den Anwenderprogrammen.