

Zu Kap. 3.1

Auswahl von Operanden und Befehlen im Speicher durch Adressierung

Grundprinzip: 3_20

n Speicherworte (Speicherwort gleichzeitig les- und/oder schreibbarer Speicherbereich mit m bit (m typ 8, 16, 32, 64, 128, ...))

logisch: n * Register für m bit

n typ: 1... 16 G (giga), evtl. auch kleiner

n wird binär kodiert:

$n = (0 \dots 2^k)$

k für 16 G = 2^{34} -> für die binäre Kodierung notwendig 34 binäre Adresssignale

32 bit Betriebssystem nur 3 ... 4 Gbyte adressierbar

Prozessor erzeugt genau eine konkrete binäre Belegung auf dem Adressbus -> im Speicher (bzw. Ein-/Ausgabe) wird genau ein Speicherwort angesprochen (zum Lesen bzw. evtl. auch zum Schreiben)

3.2. Prozessorgrundstruktur

3_50: elementare Rechnerarchitektur, die alle Funktionen der Befehlsausführung, Operandenmanipulation, Befehls- und Operandenauswahl ermöglicht, aber noch nicht mehr.

➔ Dieses Prinzip ist aber Grundlage aller heutigen konventionelle Rechnerarchitekturen

3.2.1. Erläuterung der Blöcke

ALE: Arithmetik-Logik-Einheit (Arithmetic Logic Unit, ALU) -> Verknüpfung von bis zu 2 E-Operanden zu 1 Ergebnis und Ergebnisstatus in Abhängigkeit des aktuellen Befehls

ALU ist komplexe kombinatorische Logik (siehe Teil RO)

→ Notwendig Register für E- und A-Operand, Ergebnis, Ergebnisstatus

OR1/2: vor der Operation die beiden E-Operanden, nach der Operation hat OR1 das Ergebnis, der 1. E-Operand wird überschrieben.

(Akkumulatorprinzip, Ergebnis kann sofort als E-Operand der nächsten Operation benutzt werden)

PSR (Prozessorstatusregister, Flagregister)

Speichert des Ergebnisstatus (bitweise), durch weitere Ergebnis-unabhängige bits den vollständigen Prozessorstatus

Ergebnisstatus-bit:

Zero (z): Nullergebnis

Sign (s): Vorzeichen des Ergebnisses

Carry (cy): „normaler“ Überlauf

Overflow (o): weiterer Überlauf

BR (Befehlsregister) aktuell in Bearbeitung befindlicher Befehl als Binärkode

Bsp:

Addition: 000...000B

Subtraktion: 000...001B

Log. Und: 000...010B

Log. Oder: 000...011B

...

Je Befehlstyp genau eine nichtvorzeichenbehaftete Binärzahl.

BD (Befehlsdekoder): Erzeugt aus dem aktuellen kodierten Befehl im Befehlsregister durch Dekodierung Steuersignale für die ALE und die Befehlsvariante für die AST (kombinatorische Logik)T

AST (Ablaufsteuerung):

Steuert den logisch-zeitlichen Ablauf des aktuellen Befehls im Prozessor in Abhängigkeit der vom BD bereitgestellten Befehlsvariante (BV). Befehlgruppen haben unterschiedliche BV (nicht unbedingt die Gruppen, die im Weiteren zusammengefasst werden), unterschiedliche BV haben unterschiedliches logisch-zeitliches Verhalten).

- ➔ AST ist i.a. ein synchroner sequentieller Automat, Takt ist zumeist der Prozessortakt
- ➔ Die AST hat logische Signalverbindungen zu allen übrigen Funktionseinheiten im Prozessor (hier nicht dargestellt, nur die dargestellt die von und nach außen gehen, über den Steuerbus)

BA (Befehlsadressregister, Befehlszähler, Program Counter, PC)

Enthält die Adresse des aktuellen und sequentiell auch die des darauf folgenden Befehls)

- ➔ Veränderbar über $BA := BA + 1$ oder Überschreiben mit neuem Wert

OA (Operandenadressregister)

Enthält die Adresse eines aktuellen Operanden.

- ➔ Veränderbar über Überschreiben mit neuem Wert

DT, AT, ST (Daten-, Adress-, Steuertreiber, keine logische Funktion, elektronische Verstärker, siehe Kap.2)

- ➔ Diese Grundstruktur ist vorgesehen für Princeton-Architektur (ein Systembus (Datenbus, Adressbus, Steuerbus zu Befehls-, Operandenspeicher und Ein-/Ausgabe)

Im Weiteren i.a. Princetonarchitektur in diesem Fach.

3.2.2 Ablauf von befehlen in der Grundstruktur

Befehl als Beispiel: ALE-Verknüpfung:

1. Operand: OR1 (Bsp. OR1=2H)
2. Operand wird vom Speicher nach OR1 transportiert, Adresse des Speicherwortes ist in OA (Bsp. OA=200H)
3. Ergebnis ist in OR1
4. Adresse des aktuellen Befehls ist in BA (Bsp. BA=100H)
5. Am Ende des Befehls ist die Adresse des nächstfolgenden Befehls in BA

Im Programmspeicher auf Adresse 100H Bsp. für Subtraktion 03H als Binärcode

Im Operandenspeicher auf Adresse 200H Bsp. 2H

Weiter in der nächsten VL mit Ablauf zu 3_50.