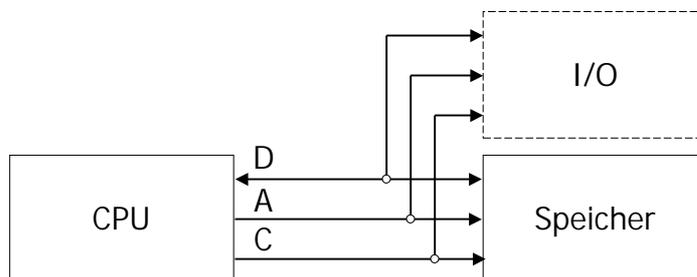


13.3 Datenpfade zur Befehlsdurchführung

Bernd Becker – Technische Informatik II

Grober Aufbau der RE-TI-II



CPU, Memory und ihr Zusammenspiel

- 3 Busse:
 - 32 Bit breiter Datenbus $D = D[31,0]$
 - 32 Bit breiter Adressbus $A = A[31,0]$
 - Kontrollbus C (Breite später festgelegt)
- 3-Bus-Architektur

- I/O-Einheiten sind wie Memory an CPU angeschlossen (später behandelt)

BB - TI II 13.3/3

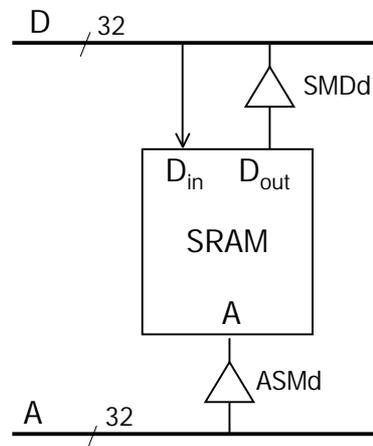
CPU, Memory und ihr Zusammenspiel (ff)

- 2 sich abwechselnde Phasen der CPU:
 - Fetch-Phase:
Lädt nächsten auszuführenden Befehl aus Memory in Instruktionsregister I der CPU
 - Execute-Phase:
Befehl, der in I steht, wird ausgeführt

Definition der Verbindungen zwischen den einzelnen Komponenten (=Datenpfade) durch Diagramme.

BB - TI II 13.3/4

Speicher SM - graphisch



BB - TI II 13.3/5

Speicher = statisches RAM *SM*

Datenein- und ausgänge mit Datenbus D verbunden,
Adressleitungen mit Adressbus A verbunden.

Treiber $ASMd$ immer enabled

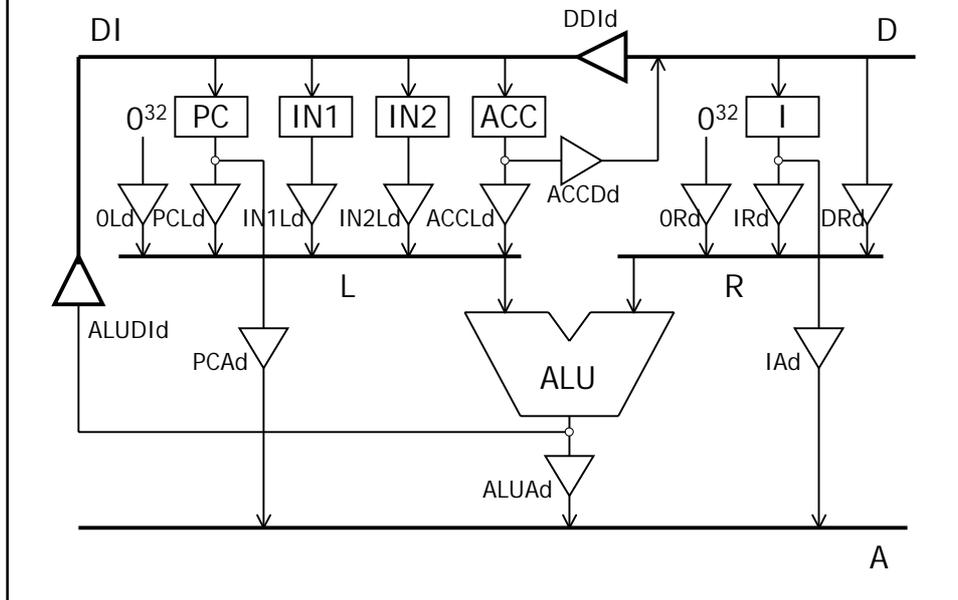
→ Reduktion des Fanout auf A-Bus.

(Namenskonvention für Treiber:

Treiber zwischen Bus/Baustein X und Bus Y: XYd)

BB - TI II 13.3/6

Datenpfade der CPU - graphisch



Datenpfade der CPU

CPU besteht aus:

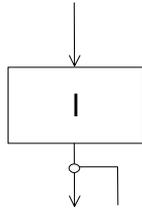
- Zähler PC
- 3 für Benutzer sichtbare Register ACC, IN1, IN2
und Instruktionsregister I
- ALU
- CPU-interne Busse
 - L, R für linken bzw. rechten Operanden der ALU
 - interner Datenbus DI

Register, PC, ALU, Busse und Treiber sind 32 Bit breit.

OLd und ORd können O^{32} auf L bzw. R legen.

Datenpfade der CPU (ff)

Verfeinerung des Diagramms an Stelle

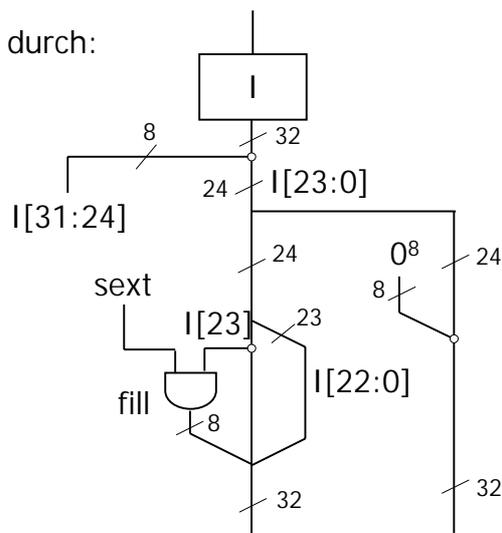


Bits $I[31,24]$ sollen nie auf R bzw. A gelegt werden.
Auf A soll $0^8I[23:0]$ gelegt werden und auf R je nach
Operation der ALU $0^8I[23:0]$ oder $\text{sext}(I[23:0])$.

BB - TI II 13.3/9

Datenpfade der CPU (ff)

Ersetze also durch:



Datenpfade der CPU (ff)

sext ist ein Kontrollsignal, das aktiv (=1) ist, falls sext(I[23:0]) gebraucht wird.

Andernfalls wird fill mit 0⁸ belegt.

BB - TI II 13.3/11

Zu zeigen:

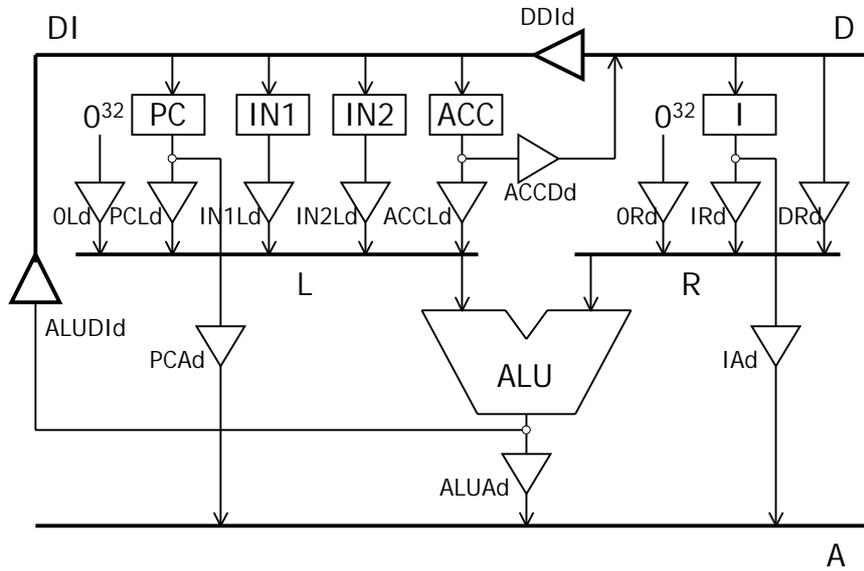
Datenpfade genügen, um Fetch-Phase und Abarbeitung aller Befehle in Execute-Phase durchzuführen.

(Danach:

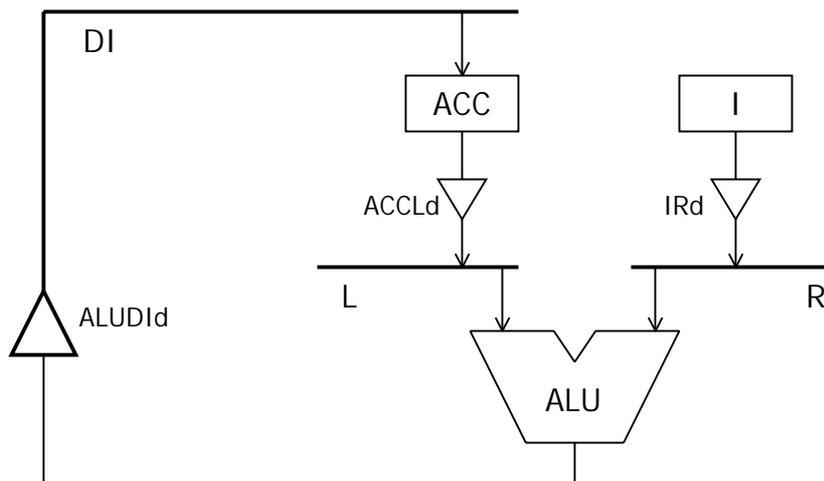
Berechne Kontrollsignale zur Steuerung der Treiber, ALU, zum Takten der Register usw.)

BB - TI II 13.3/12

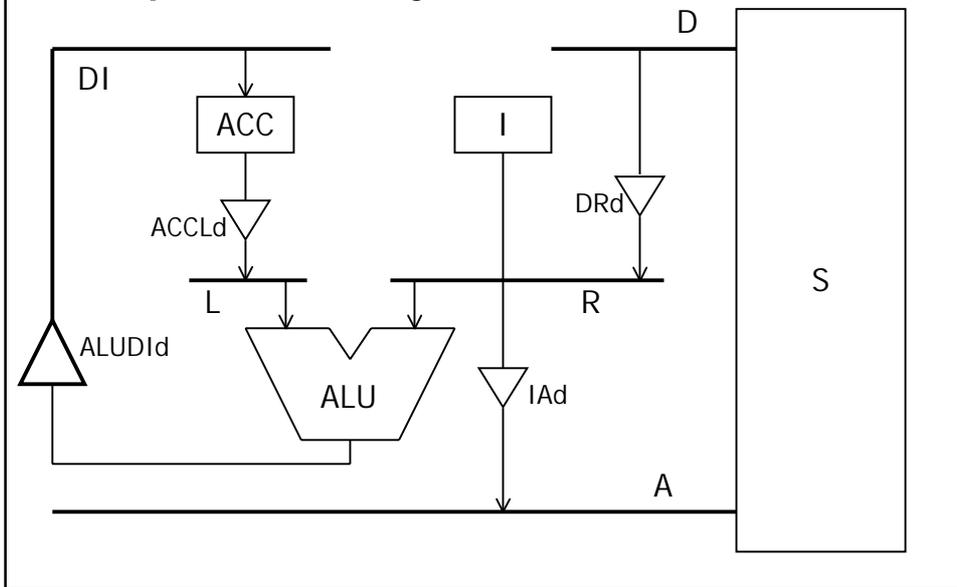
Datenpfade: Compute Immediate



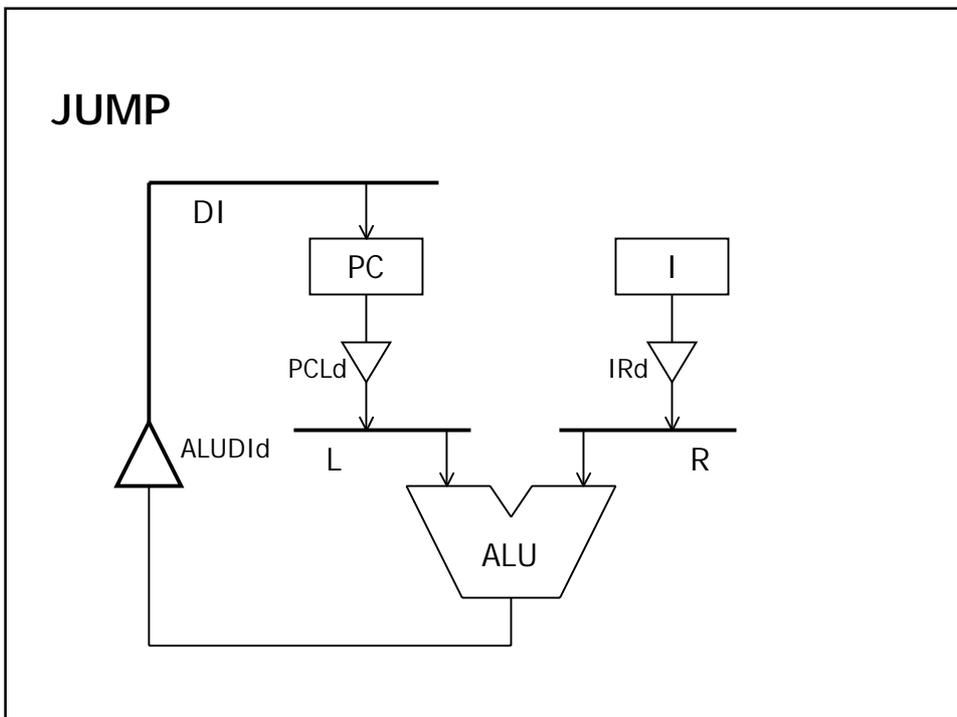
Compute immediate



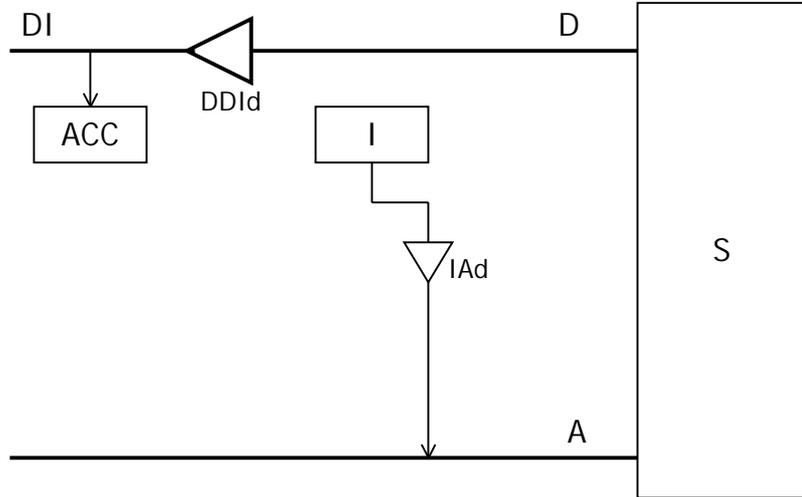
Compute memory



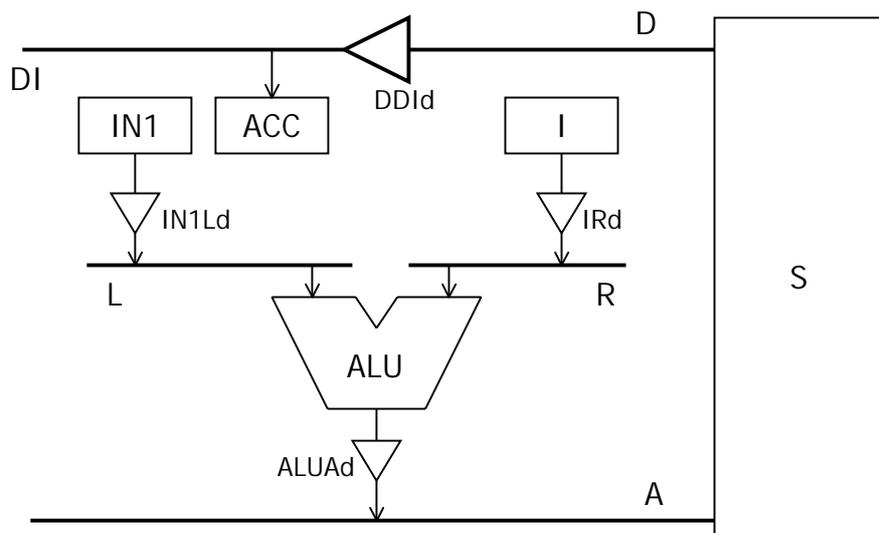
JUMP



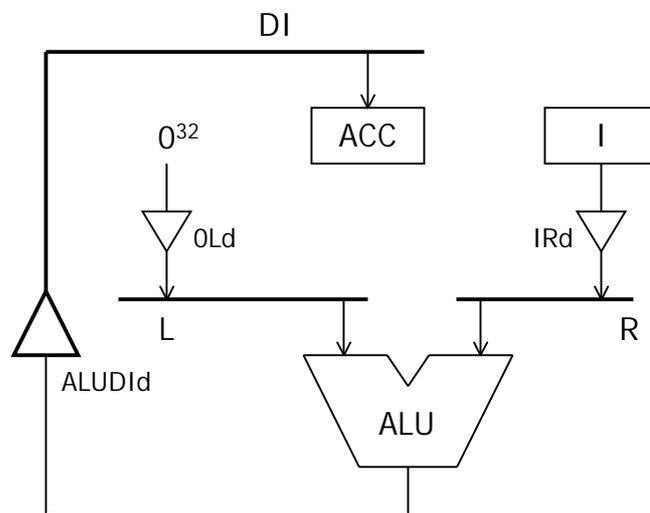
LOAD i



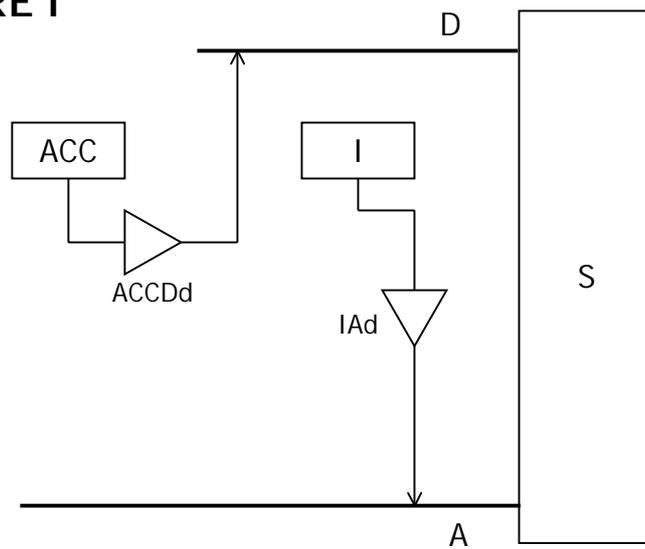
LOADIN1 i



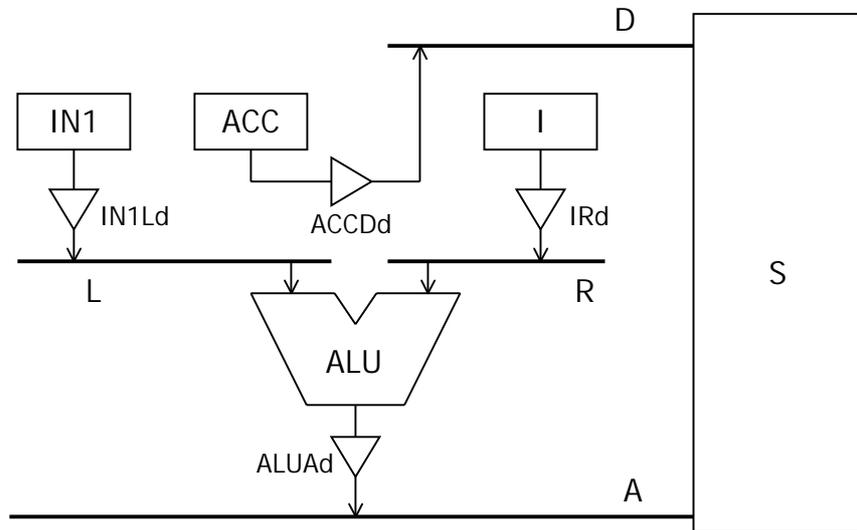
LOADI i



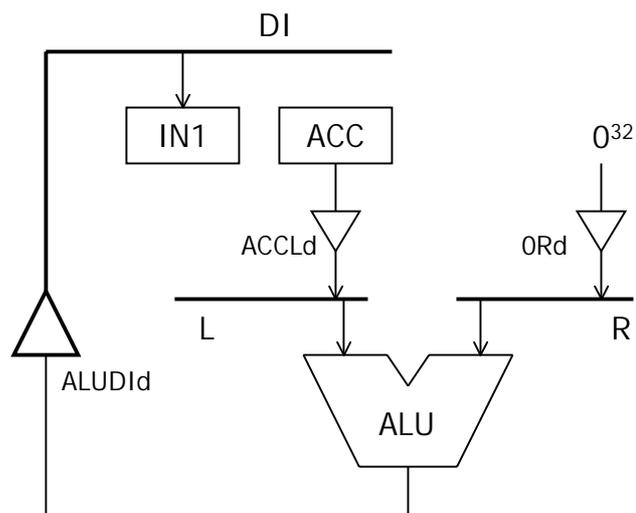
STORE i



STORE IN1 i



MOVE ACC IN1



Insgesamt gilt also

- Datenpfade genügen,
Busse müssen zur gleichen Zeit
nicht mehrfach benutzt werden.

BB - TI II 13.3/25

Zusätzliche Befehle

...die man ohne zusätzliche Hardware realisieren kann:

- Load-Befehle, bei denen die Daten in ein beliebiges Register $r \in \{PC, IN1, IN2, ACC\}$ geladen werden
 - Zur Kodierung verwende zusätzliches D-Feld bei Load-Befehlen, das bisher ungenutzt war
 - → Befehle LOAD r i
etc für $r \in \{PC, IN1, IN2, ACC\}$

BB - TI II 13.3/26

Zusätzliche Befehle (ff)

- Ebenso Compute-Befehle mit beliebigem Register als Ergebnisregister
Nutze auch hier bisher ungenutztes D-Feld

Bemerkung:

Bei Load/Compute mit Parameter $r = PC$ entfällt
 $\langle PC \rangle := \langle PC \rangle + 1$

Befehlsvorrat siehe Datenblatt

BB - TI II 13.3/27

Realisierung der Komponenten mit FAST-Bausteinen

- Register: 4 x 8-Bit-Register 74F374
- PC: 8 4-Bit-Zähler 74F163 kaskadiert
- ALU: 8 4-Bit-ALUs 74F382 kaskadiert
- Treiber:
 - Register mit Treiber auf nur 1 Bus (IR1,IR2):
Benutze interne Treiber von 74F374
 - andere Register: interne Treiber permanent enabled,
externe Treiber: 4 x 8-Bit-Treiber 74F244

BB - TI II 13.3/28

Realisierung der Komponenten mit FAST-Bausteinen (ff)

- Speicher SM:

CY7C191 = $2^{16} \times 4$ -SRAM

Um Wortbreite 32 zu erreichen: 8 x CY7C191

→ vorläufig nur Zellen mit Adresse $\in \{0, \dots, 2^{16}-1\}$

Bei Adresse $A \geq 2^{16}$: Ignoriere oberste 16 Adressbits,
d.h. Zugriff auf Adresse $A \bmod 2^{16}$

BB - TI II 13.3/29

Zu generierende Kontrollsignale:

- Clocksignale für alle Register r , Bez.: rck
- Output enable Signale (active low) für alle Treiber XYd , Bez.: $/XYd\bar{o}e$
- Funktions-Select-Signale $f[2:0]$ zum Selektieren der Funktion, die von ALU ausgeführt wird

BB - TI II 13.3/30

Zu generierende Kontrollsignale (ff)

- Eingangübertrag c_{in} der ALU
- Signale $\overline{PC\ clear}$, $\overline{PC\ load}$ für PC
- sext zur Berechnung der Füllbits bei 24-Bit-Immediate-Konstanten
- Schreibsignal (active low) \overline{SMw} für Speicher