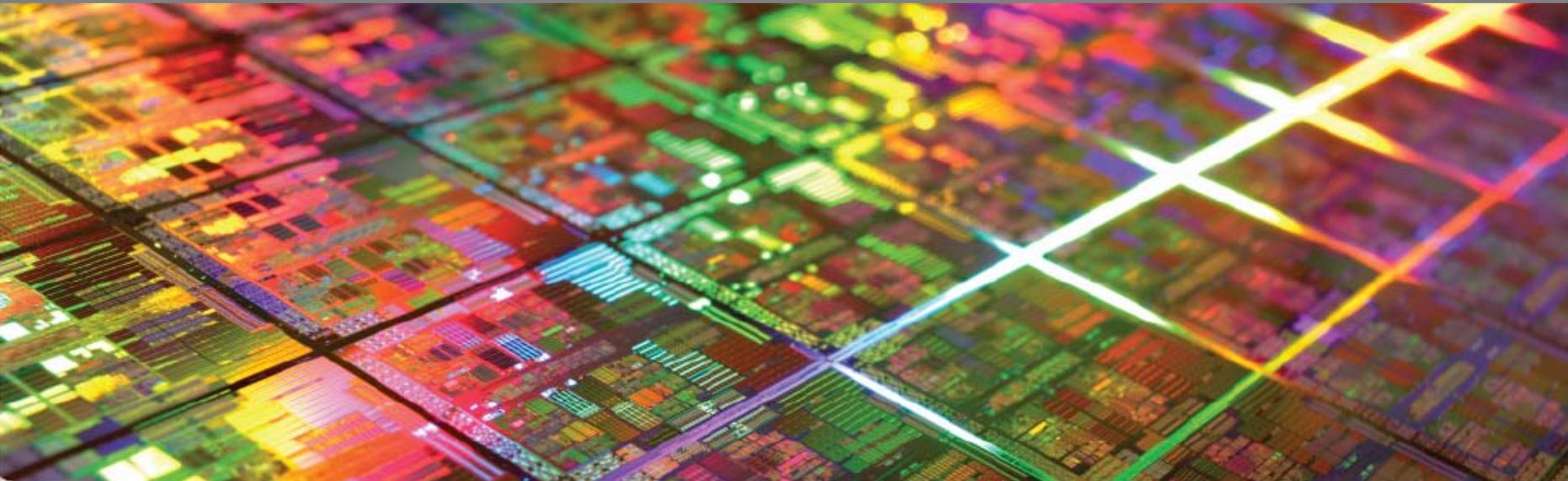


Rechnerstrukturen

Vorlesung im Sommersemester 2010

Prof. Dr. Wolfgang Karl

Fakultät für Informatik – Lehrstuhl für Rechnerarchitektur und Parallelverarbeitung



Vorlesung Rechnerstrukturen

- **Kapitel 2: Parallelismus auf Maschinenbefehlsebene**
- 2.1 Nebenläufigkeit

Parallelismus auf Maschinenbefehlsebene

■ RISC (Reduced Instruction Set Computers)

■ Einschränkungen skalarer Pipelines

■ Obere Grenze des Durchsatzes:

- $IPC \leq 1$ oder $CPI \geq 1$

■ Ineffiziente Pipeline

- Lange Latenzzeiten für eine Instruktion

■ Pipeline Stall Strategie

- Anhalten der Pipeline bewirkt, dass nachfolgende Befehle ebenfalls warten müssen

Nachfolgende Befehle müssen warten
(Backward Propagation of Stalling)



Wartende
Instruktion

Überholen der wartenden
Instruktion nicht erlaubt!

Parallelismus auf Maschinenbefehlsebene

- **RISC (Reduced Instruction Set Computers)**
- Einschränkungen skalarer Pipelines
 - Obere Grenze des Durchsatzes:
 - $IPC \leq 1$ oder $CPI \geq 1$
 - Lösung: Nebenläufigkeit, parallele Pipelines
 - Ineffiziente Pipeline
 - Lange Latenzzeiten für eine Instruktion
 - Lösung: diversifizierte, spezialisierte Pipelines
 - Pipeline Stall Strategie
 - Anhalten der Pipeline bewirkt, dass nachfolgende Befehle ebenfalls warten müssen
 - „Out-of-Order“ Strategie, verteilte Ausführungspipelines

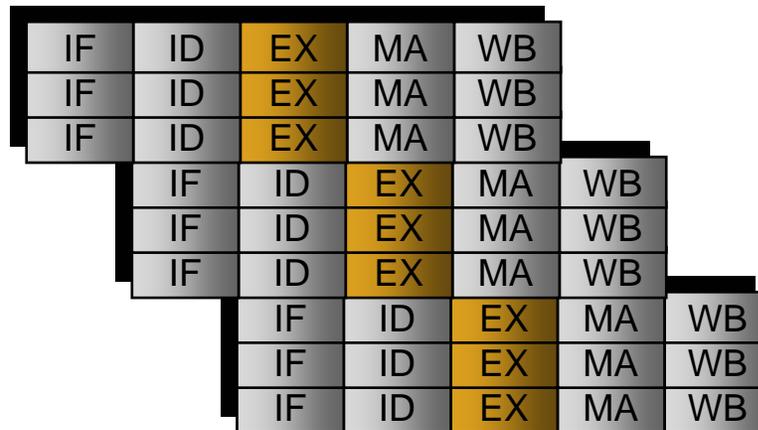
Parallelismus auf Maschinenbefehlsebene

■ Nebenläufigkeit

- Dynamische Ansätze
 - Superskalartechnik
- Statische Ansätze
 - VLIW (Very Long Instruction Word)
 - EPIC (Explicitly Parallel Instruction Computer)

Parallelismus auf Maschinenbefehlsebene

- Nebenläufigkeit
- Dynamische Ansätze
 - Superskalartechnik
 - Anstoßen (Issue) von n Befehlen pro Zyklus
 - Max. IPC = n Befehle pro Zyklus



Parallelismus auf Maschinenbefehlsebene

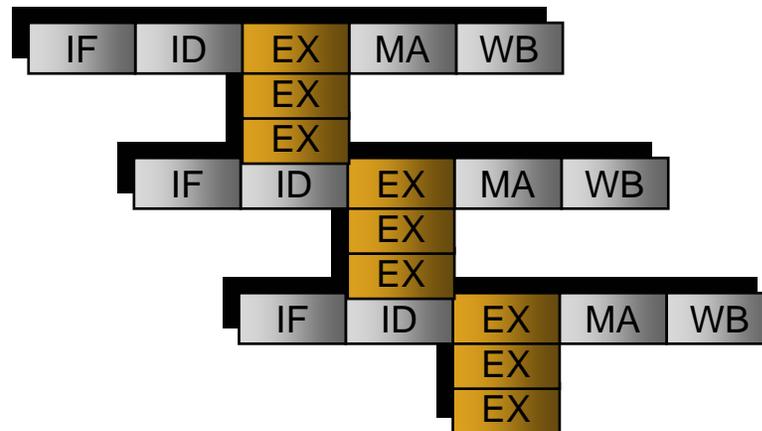
- Nebenläufigkeit

- Statische Ansätze

- VLIW

- Anstoßen (Issue) von n Operationen pro Zyklus

- Max. IPC = n Operationen pro Zyklus = 1 VLIW Befehl pro Zyklus



Parallelismus auf Maschinenbefehlsebene

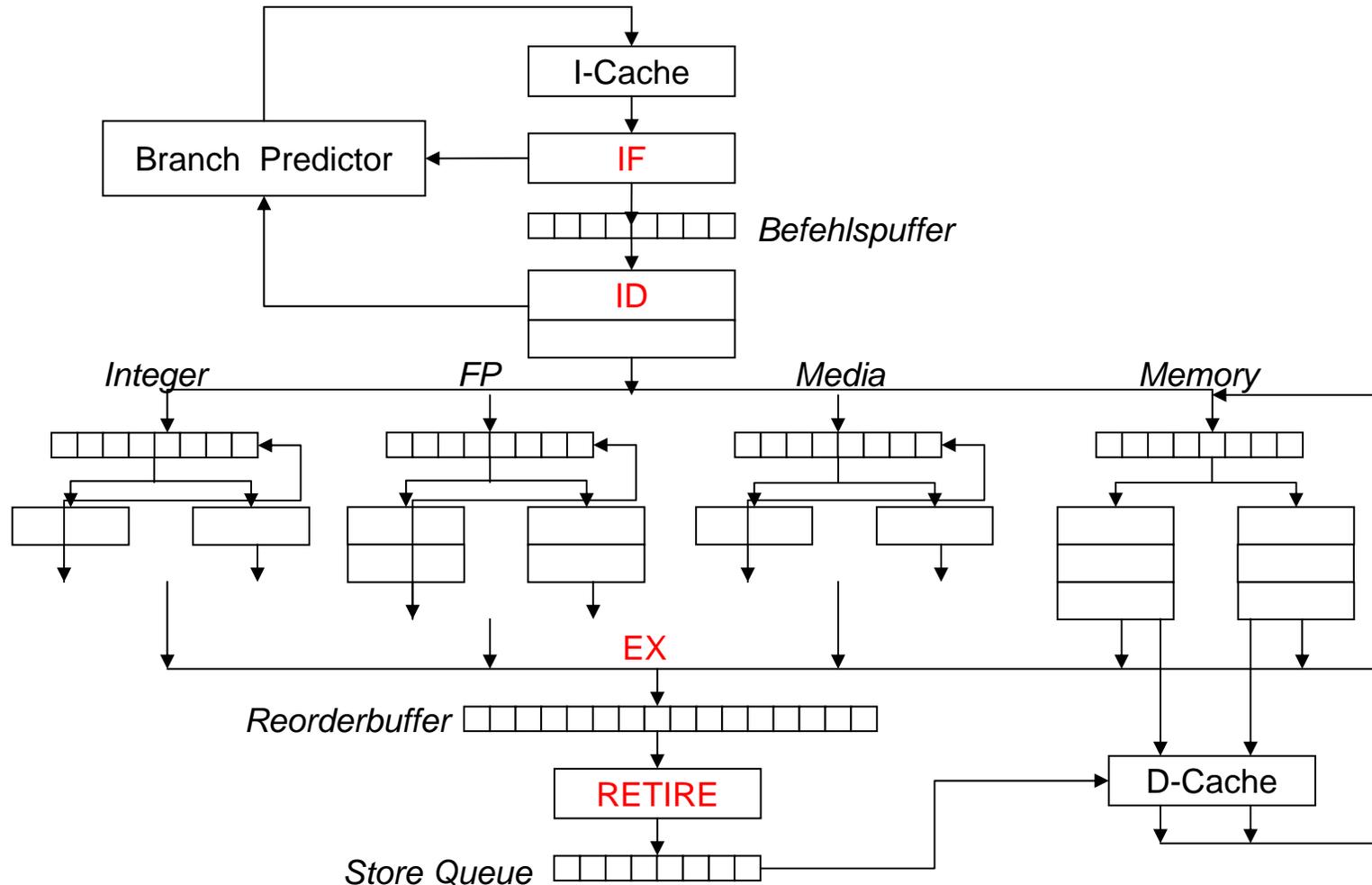
- **Nebenläufigkeit**
- RISC → Superskalar
 - Mehrfachzuweisungsmethoden (multiple issue)
 - Die Superskalar-Technik ermöglicht es, pro Takt mehrere Befehle den Ausführungseinheiten zuzuordnen und eine gleiche Anzahl von Befehlsausführungen pro Takt zu beenden.
 - Superskalare RISC-Prozessoren:
 - RISC-Charakteristika werden auch heute noch weitgehend beibehalten
 - Lade-/Speicher-Architektur
 - Festes Befehlsformat (z. B.: Befehlslänge: 32 Bit)
 - Entwurfsziel: Erhöhung des IPC (Instruction per Cycle)
 - Heutige Mikroprozessoren nutzen Befehlsebenenparallelität durch die Pipelining- und Superskalartechnik

Parallelismus auf Maschinenbefehlsebene

- Nebenläufigkeit
- Superskalarer Prozessor
 - Nützt den Parallelismus auf Befehlsebene aus
 - Vielstufige Befehlspipeline
 - Superskalartechnik
 - Eigenschaften:
 - Mehrere voneinander unabhängige Ausführungseinheiten
 - Zur Laufzeit werden pro Takt mehrere Befehle aus einem sequentiellen Befehlsstrom den Verarbeitungseinheiten zugeordnet und ausgeführt
 - Dynamische Erkennung und Auflösung von Konflikten zwischen Befehlen im Befehlsstrom ist Aufgabe der Hardware

Superskalartechnik

■ Superskalarer Prozessor



Superskalartechnik

■ Superskalarer Prozessor

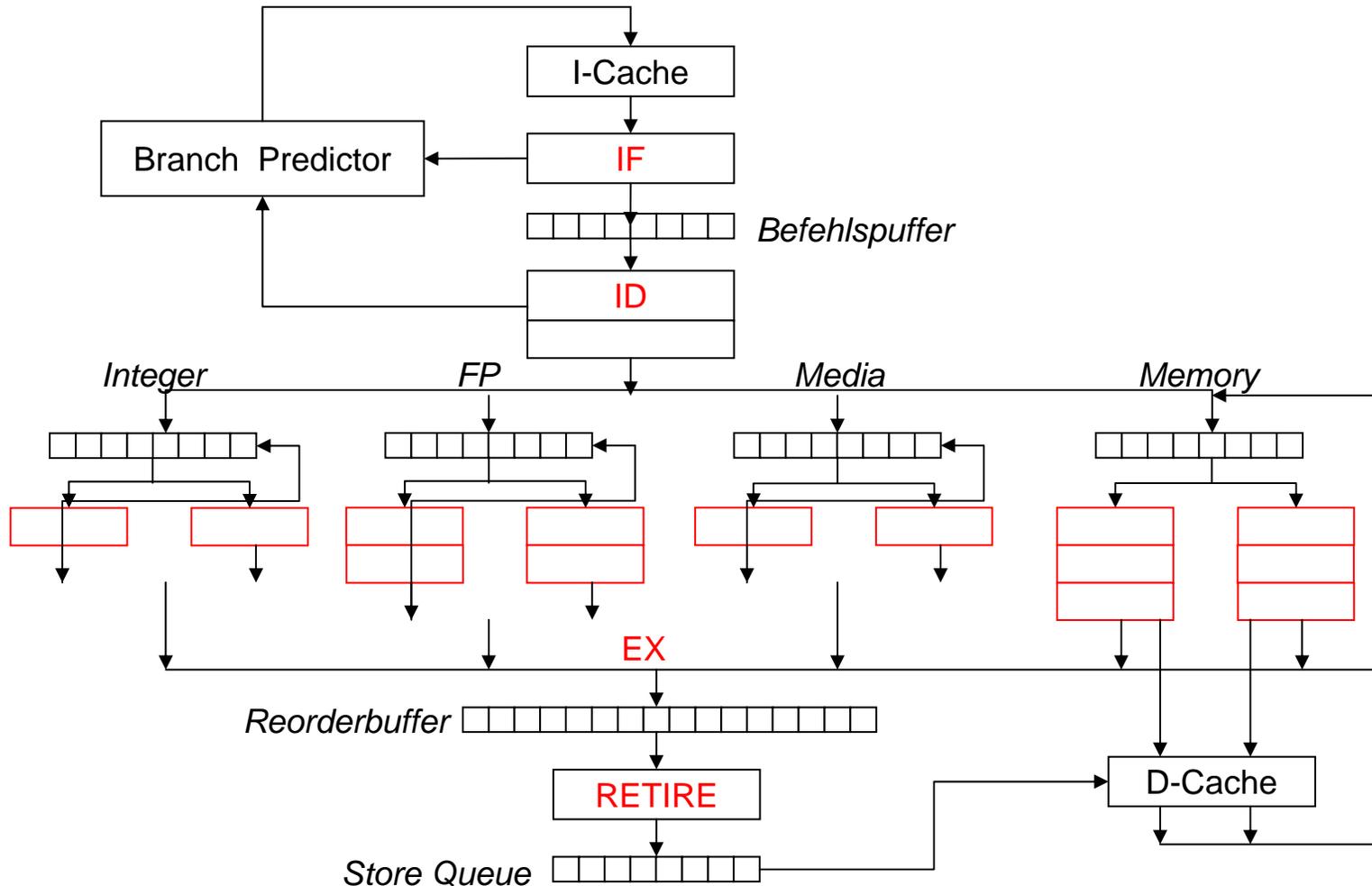
■ Komponenten

- Befehlsholeinheit (Instruction Fetch)
- Dekodiereinheit (Instruction Decode) mit Registerumbenennung (register renaming)
- Zuordnungseinheit (Instruction Issue)
- Unabhängige Verarbeitungseinheiten (Functional Units)
- Rückordnungseinheit (Retire Unit)
- Register:
 - Allzweckregister
 - Multimediaregister
 - Spezialregister

- *Anmerkung: Die Bezeichnungen der Einheiten sind bei den verschiedenen Prozessoren nicht einheitlich!*

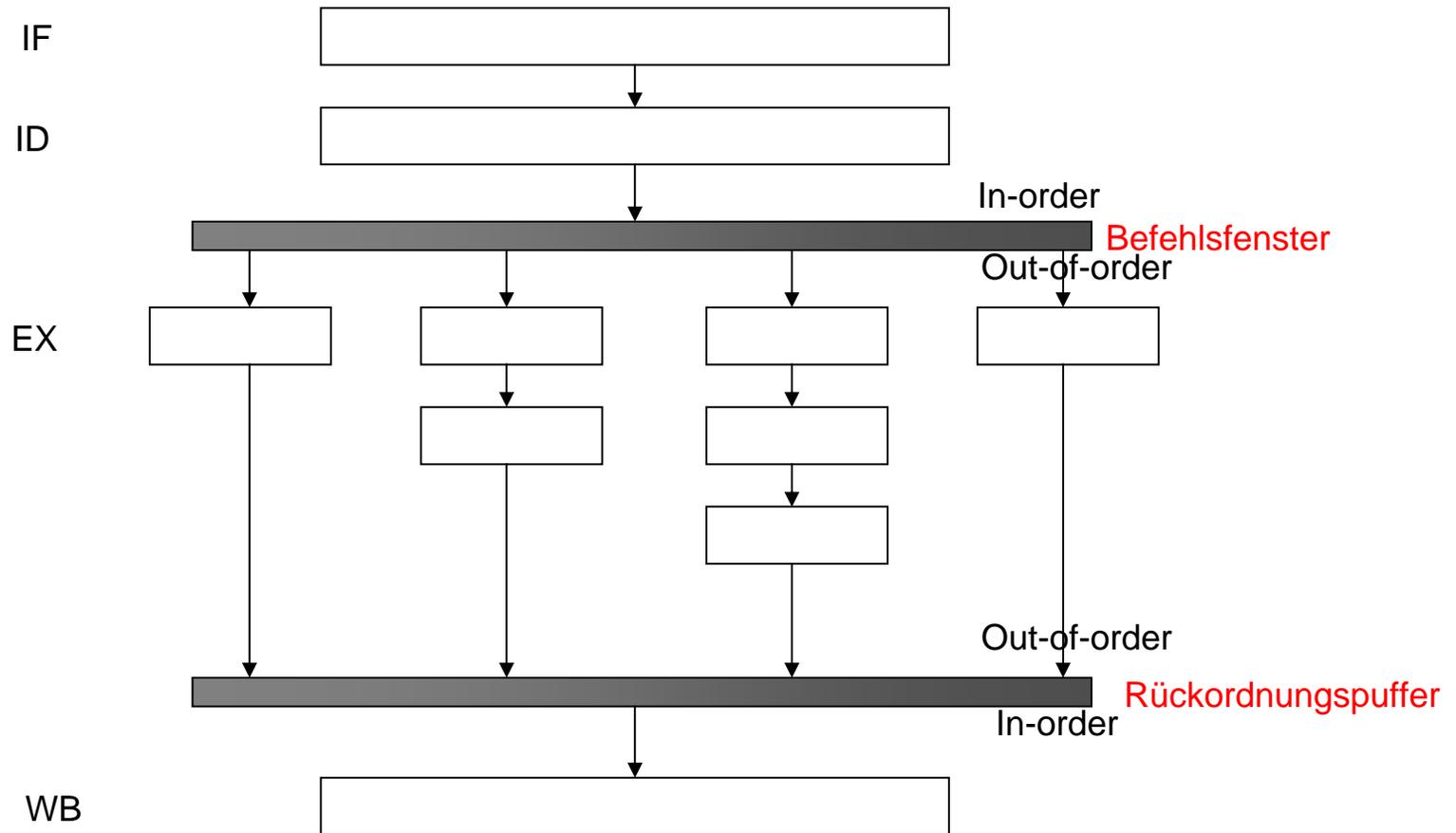
Superskalartechnik

■ Superskalarer Prozessor: spezialisierte Pipelines



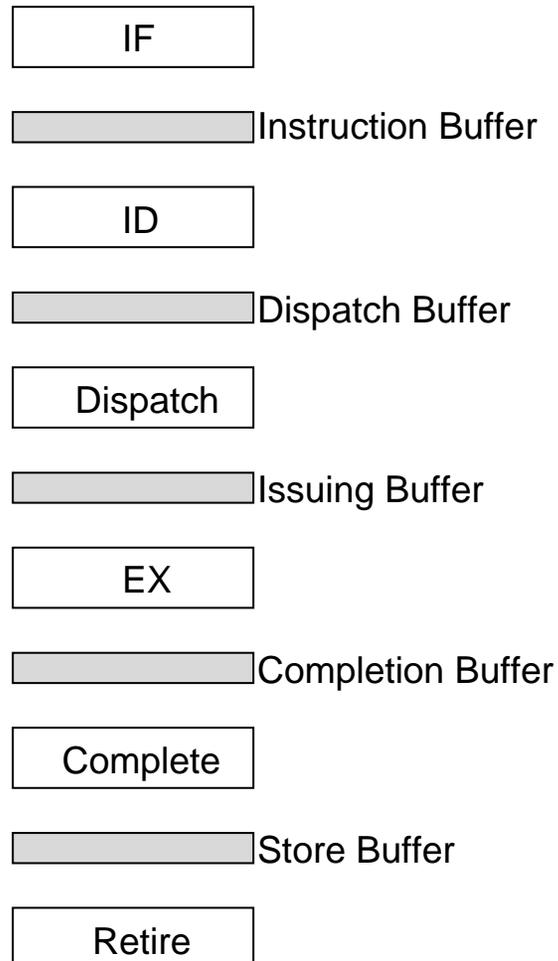
Superskalartechnik

■ Superskalarer Prozessor: dynamische Pipelines

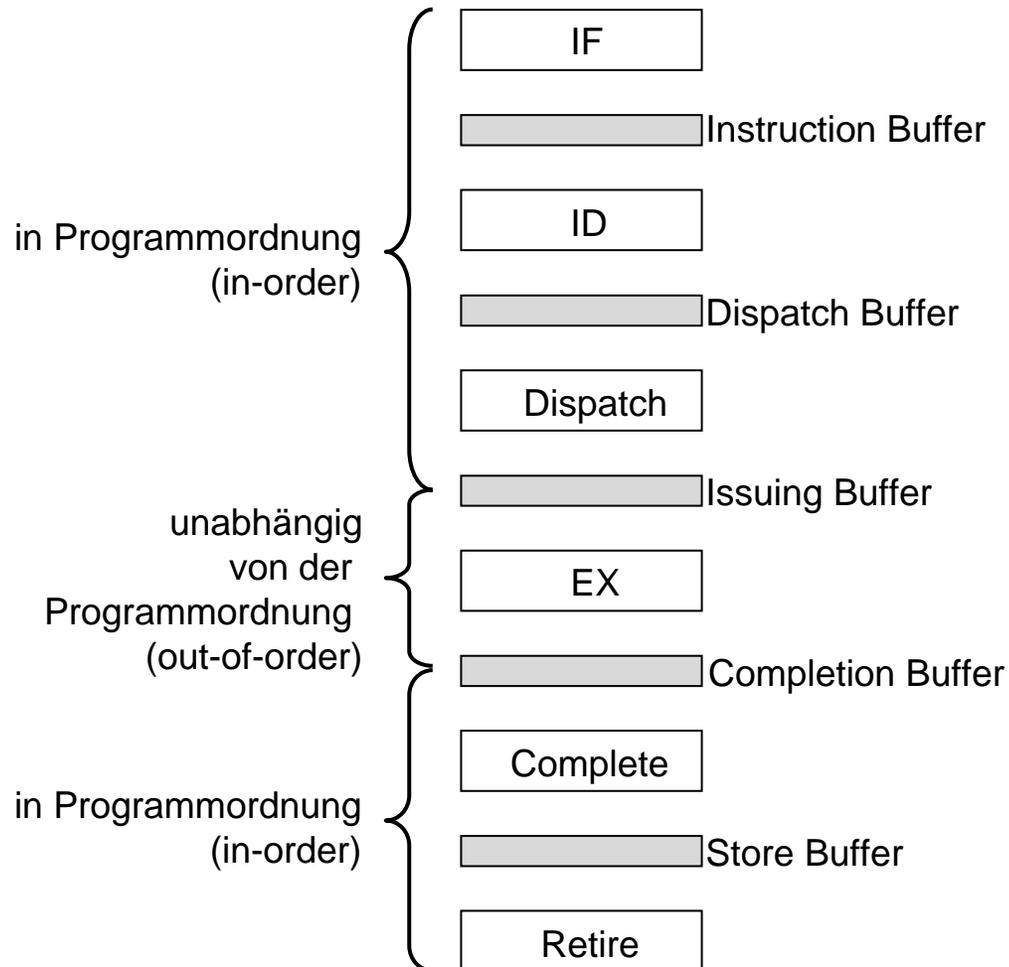


Superskalartechnik

■ Superskalare Prozessorphipeline



■ Superskalare Prozessorpipeline



Superskalartechnik

■ Superskalare Prozessorpipeline

■ 1. In-order-Abschnitt

- Befehle werden entsprechend ihrer Programmordnung bearbeitet
- Umfasst
 - die Befehlsholphase (IF)
 - die Dekodierphase (ID)
 - die Zuordnungsstufe (Dispatch)
 - Dynamische Zuordnung der Befehle an die Ausführungseinheiten
 - Scheduler bestimmt die Anzahl der Befehle, die im nächsten Takt zugeordnet werden können

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Out-of-order-Abschnitt
 - Ausführungsphase

- 2. In-order-Abschnitt
 - Gültigmachen der Ergebnisse entsprechend der ursprünglichen Programmordnung
 - Erhalten der korrekten Programmsemantik
 - Ausnahmeverarbeitung (precise interrupts)
 - Spekulation

Superskalartechnik

- **Superskalare Prozessorpipeline**
- **Befehlsholphase (IF Phase)**
 - **Befehlsbereitstellung**
 - Holen mehrerer Befehle aus dem Befehls-Cache in den Befehlsholpuffer
 - Anzahl der Befehle, die geholt werden, entspricht typischer Weise der Zuordnungsbandbreite
 - Welche Befehle geholt werden hängt von der Sprungvorhersage ab
 - **Verzweigungseinheit**
 - Überwacht die Ausführung von Verzweigungen, Sprungbefehlen
 - Spekulatives Holen von Befehlen
 - Spekulation über weiteren Programmverlauf wird von dynamischen Sprungvorhersagetechnik entschieden
 - Verwendung der Vorgeschichte von Sprüngen
 - Gewährleistet im Falle einer Fehlspekulation die Abänderung der Tabellen sowie das Rückrollen der fälschlicherweise ausgeführten Befehle
 - **Befehlsholpuffer**
 - entkoppelt die IF Phase von der ID Phase

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Befehlsholphase (IF Phase)
 - Sprungvorhersage und spekulative Ausführung
 - Problem
 - Hohe Zuordnungs- und Ausführungsbandbreite
 - Etwa jeder 5.- 7. Befehl ist bedingter Sprungbefehl, der den kontinuierlichen Befehlsfluss in der Pipeline unterbrechen kann
 - Unter Berücksichtigung der spekulativen Ausführung von Befehlen können sich mehrere Sprungbefehle in der Pipeline befinden

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Beim Auftreten einer Verzweigung: Vorhersage des Sprungziels
 - Füllen der Verzögerungsphasen spekulativ mit Befehlen, die dem Sprung folgen oder die am Sprungziel stehen
 - Nach Auswertung der Sprungbedingung:
 - Fortfahren mit der Ausführung ohne Verzögerung bei korrekter Vorhersage.
 - Verwerfen der geholten Befehle bei falscher Vorhersage

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Statische Sprungvorhersage
 - Die Richtung der Vorhersage ist für einen Befehl immer gleich
 - Dynamische Sprungvorhersage
 - Die Vorhersage hängt von der Vorgeschichte ab.

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Statische Sprungvorhersage
 - Die Richtung der Vorhersage ist für einen Befehl immer gleich
 - Sprungvorhersage im Prozessor fest verdrahtet:
 - Branch-taken: Annahme, dass die Verzweigung immer stattfindet.
 - Branch-not-taken: Annahme, dass die Verzweigung nicht stattfindet.

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Statische Sprungvorhersage
 - Die Richtung der Vorhersage ist für einen Befehl immer gleich
 - Sprungvorhersage im Prozessor fest verdrahtet:
 - Branch-taken: Annahme, dass die Verzweigung immer stattfindet.
 - Branch-not-taken: Annahme, dass die Verzweigung nicht stattfindet.

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Dynamische Vorhersage
 - Berücksichtigung des Programmverhaltens
 - Die Verzweigungsrichtung hängt von der Vorgeschichte der Verzweigung ab
 - Genauere Vorhersage möglich
 - Hoher Hardware-Aufwand!

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Sprungziel-Cache: Branch Target Address Cache (BTAC), Branch Target Buffer (BTB)
 - Speichert die Adresse der Verzweigung und das entsprechende Sprungziel

Adresse der Verzweigung	Sprungziel-adresse

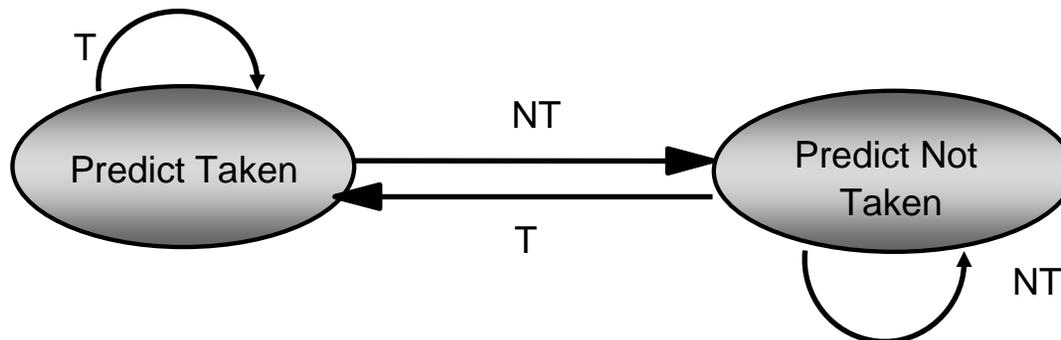
Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Sprungziel-Cache: Branch Target Address Cache (BTAC), Branch Target Buffer (BTB)
 - Sprungverlaufstabelle, Branch History Table (BHT)
 - Festhalten des Verhaltens der Sprungbefehle während der Ausführung des Programms: Prädiktoren
 - Vorhersage des Verhaltens eines geholten Sprungbefehls

Adresse der Verzweigung	Sprungziel-adresse	Vorher-sagebits

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Branch Prediction Buffer, Branch History Table
 - Vorhersagebit:
 - Wenn das Bit gesetzt ist, wird angenommen, dass der Sprung ausgeführt wird.
 - Wenn das Bit nicht gesetzt ist, wird angenommen, dass der Sprung nicht ausgeführt wird.
 - Bei einer Fehlannahme: Invertieren des Bits

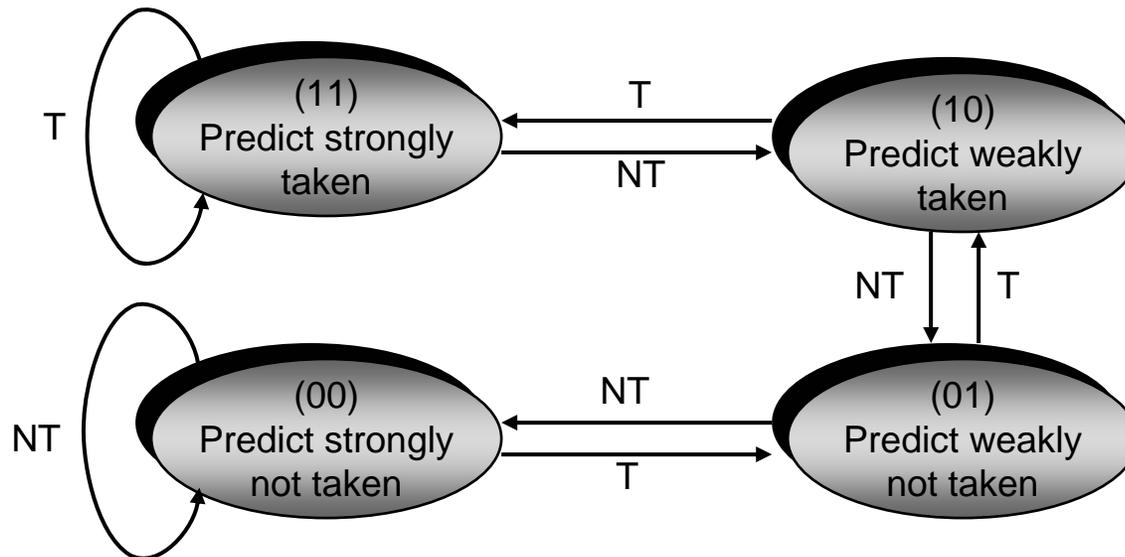


Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Branch Prediction Buffer, Branch History Table: Zwei-Bit Predictor
 - Zwei Bit pro Eintrag für die Kodierung der Vorhersage → vier Zustände:
 - Sicher genommen (strongly taken)
 - Vielleicht genommen (weakly taken)
 - Vielleicht nicht genommen (weakly not taken)
 - Sicher nicht genommen (strongly not taken)
 - In einem sicheren Zustand sind zwei aufeinander folgende Fehlannahmen notwendig, um die Vorhersageannahme umzudrehen.

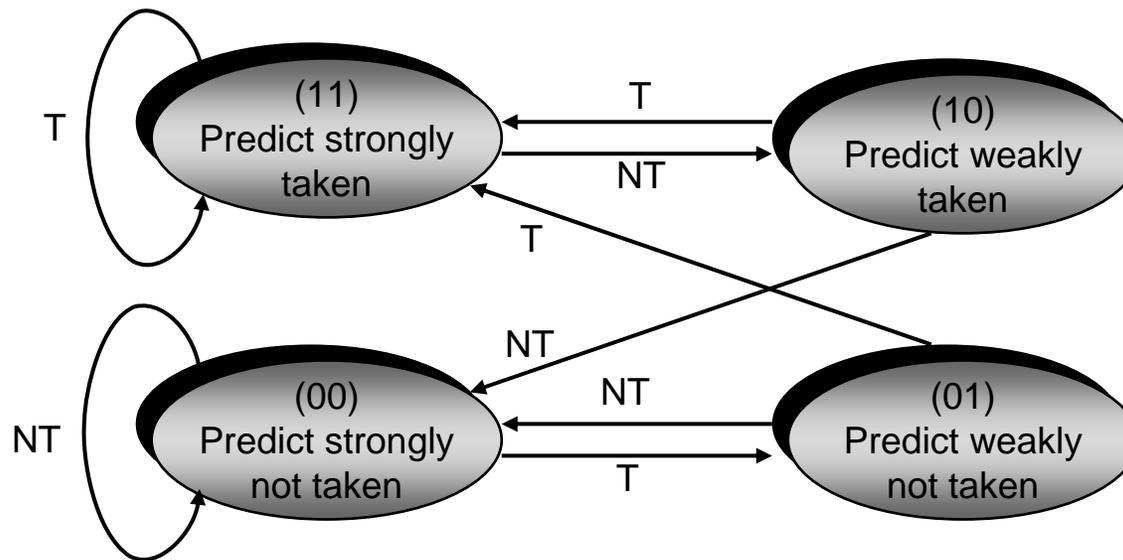
Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Branch Prediction Buffer, Branch History Table: Zwei-Bit Predictor mit Sättigungszähler (Two Bit Predictor with Saturation Scheme)



Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Branch Prediction Buffer, Branch History Table: Zwei-Bit Predictor mit Hystereseemethode (Two Bit Predictor with Hysteresis Scheme)



Superskalartechnik

- **Superskalare Prozessorpipeline**
- Sprungvorhersage (Branch Prediction): Vorhersage des Verhaltens bei Verzweigungen
 - Dynamische Sprungvorhersagetechniken
 - Sehr aufwendige Techniken für superskalare Prozessoren für die Gewährleistung einer möglichst genauen Vorhersage
 - (m,n)-Korrelationsprädiktoren
 - Zweistufige adaptive Prädiktoren
 - Gselect- und gshare-Prädiktoren
 - Hybridprädiktoren
 - Literatur zur Sprungvorhersage:
 - Brinkschulte/Ungerer: Microcontroller und Mikroprozessoren: Kap. 2.4.6, 7.2

Superskalartechnik

■ Superskalare Prozessorpipeline

■ Dekodierphase (ID Phase)

■ Dekodierung der im Befehlspuffer abgelegten Befehle

- Anzahl der Befehle, die dekodiert werden, entspricht typischer Weise der Befehlsbereitstellungsbandbreite

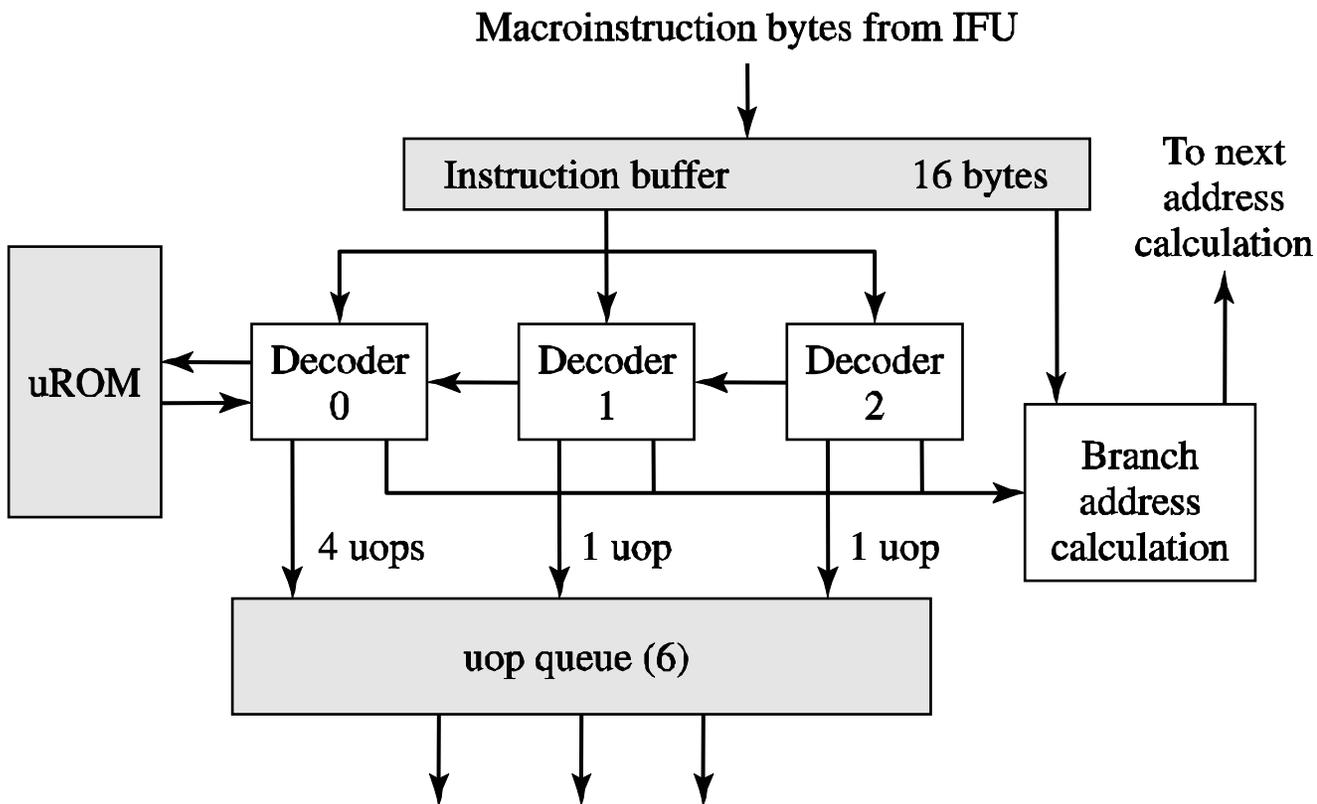
■ Bei CISC-Architekturen (IA-32)

■ Aufteilung der Dekodierung in mehrere Schritte

- Bestimmung der Grenzen der geholten Befehle
- Dekodierung der Befehle
- Generierung einer Folge von RISC-ähnlichen Operationen mit Hilfe dynamischer Übersetzungstechniken
- Ermöglicht effizientes Pipelining und superskalare Verarbeitung
- Beispiel Intel Pentium- und AMD Athlon-Familie

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Dekodierphase (ID Phase)
 - Bei CISC-Architekturen (IA-32): Intel Pentium Pro



Superskalartechnik

■ Superskalare Prozessorpipeline

■ Dekodierphase (ID Phase)

■ Registerumbenennung

- Dynamische Umbenennung der Operanden- und Resultatsregister
- Abbildung der nach außen hin sichtbaren Architekturregister in interne physikalische Register
 - Zur Laufzeit wird für jeden Befehl das jeweils spezifizierte Zielregister auf ein noch nicht belegtes physikalisches Register abgebildet
 - Nachfolgende Befehle, die dasselbe Architekturregister als Operandenregister verwenden, erhalten das entsprechende physikalische Register
 - Anzahl der Umbenennungsregister kann die Anzahl der Architekturregister überschreiten
- Auflösung von Konflikten aufgrund von Namensabhängigkeiten

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Dekodierphase (ID Phase)
 - Schreiben der Befehle in ein Befehlsfenster (instruction window)
 - Folge:
 - Befehle sind durch die Sprungvorhersage frei von Steuerflussabhängigkeiten
 - Befehle sind aufgrund der Registerumbenennung frei von Namensabhängigkeiten

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Zuordnungsphase (Dispatch)
 - Zuführung der im Befehlsfenster wartenden Befehle zu den Ausführungseinheiten
 - Dynamische Auflösung der Konflikte aufgrund von echten Datenabhängigkeiten und Ressourcenkonflikten
 - Zuordnung bis zur maximalen Zuordnungsbandbreite pro Takt

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Zuordnungsphase (Dispatch)
 - Rückordnungspuffer (reorder buffer)
 - Festhalten der ursprünglichen Befehlsanordnung
 - Eintragen der Befehle, die die Dekodierphase verlassen und in das Befehlsfenster eingetragen werden
 - Während der folgenden Phasen, die ein Befehl zu durchlaufen hat, wird dessen jeweiliger Ausführungsstand protokolliert.

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Zuordnungsphase (Dispatch)
 - Zweistufige Zuweisung:
 - Umordnungspuffer (Reservierungstabellen, reservation stations)
 - Liegen vor den Verarbeitungseinheiten
 - Jede Ausführungseinheit hat seinen eigenen Umordnungspuffer oder mehrere Ausführungseinheiten teilen sich einen Umordnungspuffer
 - Zuordnung eines Befehls an Umordnungspuffer kann nur erfolgen, wenn ein freier Platz vorhanden ist, ansonsten müssen die nachfolgenden Befehle warten (Auflösen von Ressourcenkonflikten)

Superskalartechnik

■ Superskalare Prozessorpipeline

■ Befehlsausführung

- Ausführung der im Opcode spezifizierten Operation und Speichern des Ergebnisses im Zielregister (Umbenennungsregister)
- Einzyklusoperationen
 - Ausführung benötigt einen Taktzyklus
- Mehrzyklusoperationen
 - Ausführung einer Operation auf einer Ausführungseinheit kann mehrere Zyklen dauern
 - Ausführungs-Pipeline, arithmetische Pipeline

Superskalartechnik

■ Superskalare Prozessorpipeline

■ Completion

- Eine Instruktion beendet ihre Ausführung, wenn das Ergebnis für nachfolgende Befehle bereitsteht (Forwarding, Puffer)
- Completion heißt: eine Befehlsausführung ist „vollständig“
 - Erfolgt unabhängig von der Programmordnung!
- Bereinigung der Reservierungstabellen
- Aktualisierung des Zustands des Rückordnungspuffers (Reorder Buffer)
 - Es kann eine Unterbrechung angezeigt sein.
 - Es kann ein vollständiger Befehl angezeigt werden, der von einer Spekulation abhängt.

Superskalartechnik

■ Superskalare Prozessorpipeline

■ Rückordnungsstufe (Retire)

■ Commitment:

- Nach der Vervollständigung beenden die Befehle ihre Bearbeitung (Commitment), d.h. die Befehlsresultate werden in der Programmreihenfolge gültig gemacht
- Ergebnisse werden in den Architekturregistern dauerhaft gemacht, d.h. aus den internen Umbenennungsregistern (Schattenregistern) zurück geschrieben.

■ Bedingungen für Commitment:

- Die Befehlsausführung ist vollständig
- Alle Befehle, die in der Programmordnung vor dem Befehl stehen, haben bereits ihre Bearbeitung beendet oder beenden ihre Bearbeitung im selben Takt.
- Der Befehl hängt von keiner Spekulation ab.
- Keine Unterbrechung ist vor oder während der Ausführung aufgetreten

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Rückordnungsstufe (Retire)
 - Forderung: Precise Interrupts bei Auftreten einer Unterbrechung
 - Alle Resultate von Befehlen, die in der Programmreihenfolge vor dem Ereignis stehen, werden gültig gemacht
 - Die Resultate aller nachfolgenden Befehle werden verworfen
 - Das Ergebnis des verursachenden Befehls wird in Abhängigkeit der Architektur oder der Art der Unterbrechung gültig gemacht oder verworfen, ohne weitere Auswirkungen zu haben

Superskalartechnik

- **Superskalare Prozessorpipeline**
- Rückordnungsstufe (Retire)
 - Forderung: Precise Interrupts bei Auftreten einer Unterbrechung
 - Alle Resultate von Befehlen, die in der Programmreihenfolge vor dem Ereignis stehen, werden gültig gemacht
 - Die Resultate aller nachfolgenden Befehle werden verworfen
 - Das Ergebnis des verursachenden Befehls wird in Abhängigkeit der Architektur oder der Art der Unterbrechung gültig gemacht oder verworfen, ohne weitere Auswirkungen zu haben

Superskalartechnik

■ Fallstudie: Motorola PowerPC 604

Commitment

