

Betriebssysteme

Speicherverwaltung

G. Richter

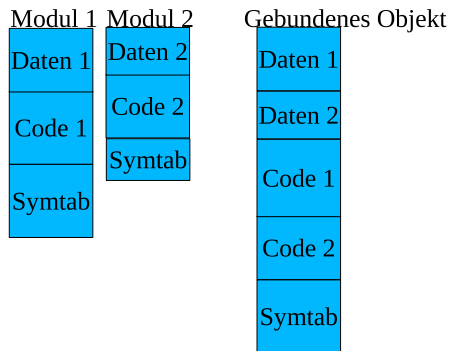
22. März 2017

Code im Speicher

Adresse	Inhalt (hex)	Assemblercode
400543:	c7 05 5f 0b 20 00 14	movl \$0x14,0x200b5f(%rip) # 6010ac
40054a:	00 00 00	
40054d:	c7 05 09 0b 20 00 1e	movl \$0x1e,0x200b09(%rip) # 601060
400554:	00 00 00	
400557:	8b 05 03 0b 20 00	mov 0x200b03(%rip),%eax # 601060
40055d:	85 c0	test %eax,%eax
40055f:	7e 14	jle 400575 <main+0x36>
400561:	8b 15 f9 0a 20 00	mov 0x200af9(%rip),%edx # 601060
400567:	8b 05 3b 0b 20 00	mov 0x200b3b(%rip),%eax # 6010a8
40056d:	01 d0	add %edx,%eax
40056f:	89 05 37 0b 20 00	mov %eax,0x200b37(%rip) # 6010ac
400575:	8b 15 e5 0a 20 00	mov 0x200ae5(%rip),%edx

Binden von Modulen

- Beim Compilieren von Modulen kennt der Compiler keine Adressen aus anderen Modulen.
- Daher werden Module immer auf die gleichen Adressen übersetzt.
- Beim Binden werden die Module in eine Datei zusammengefügt und die Referenzen neu berechnet.



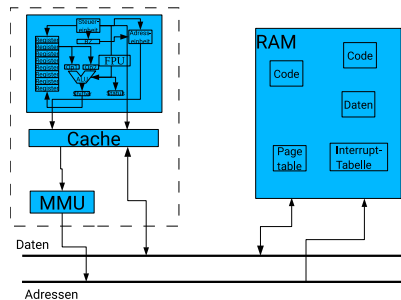
Partitionierung

- Die einfachste Art, den Speicher zu verwalten ist die Aufteilung in feste Bereiche. Dabei kommt es natürlich zu einem mehr oder weniger großen Verlust, da die Programme nicht genau so groß sind wie die Partitionen.
- Eine Verbesserung bringt die Einrichtung unterschiedlich großer Partitionen.(Bild).
- Der Aufwand beim Laden steigt, da eine passende Partition gefunden werden muss.
- Beim Laden (oder Übersetzen) müssen die Programme auf die Partition reloziert werden



Adressumrechnung, virtuelle Adressierung

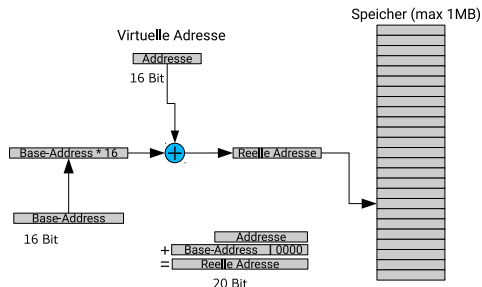
- Die von der CPU verwendete Adresse wird, bevor sie auf den Adressbus gesendet wird, von der MMU (Memory Management Unit) umgerechnet.
- Die Adresse aus der CPU wird als virtuelle Adresse bezeichnet .
- Die Adresse nach der MMU wird als reelle Adresse bezeichnet.
- Die Notwendigkeit, das Programm neu zu relocieren, entfällt.



Quelle : Wikipedia

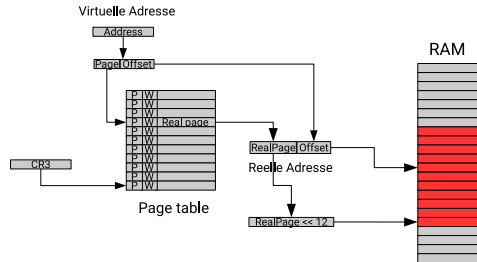
Segmentierte Adressierung

- Unterstützt dynamische Partitionierung
- Größe der Partition auf Programmgröße einstellbar
- Komplexität der Speicherverwaltung stark erhöht.
- Intel 8088
- Bietet die Möglichkeit einen größeren Adressraum als mit der Registerbreite darstellbar zu realisieren.



Virtuelle Adressierung

- Page ist Index in der Pagetable
- Offset wird aus der virtuellen Adresse übernommen
- mit P,W wird der Zugriff kontrolliert (Present, Write)
- CR3 ist Prozessorregister, das auf den Beginn der PT zeigt
- Aber: PT ist 4MB groß, da für jede Seite ein Eintrag erforderlich



Aufteilung des virtuellen Adressraums

- Daten und Code haben feste Länge
- Heap (nicht initialisierte Daten, Speicher, der während der Laufzeit angefordert wird) kann wachsen
- Stack wächst während der Laufzeit
- daher wachsen Stack und Heap aufeinander zu
- „Loch“ dazwischen

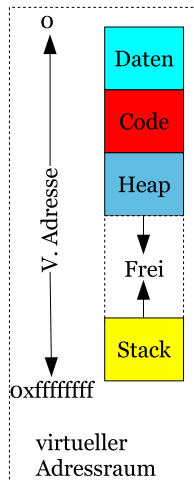
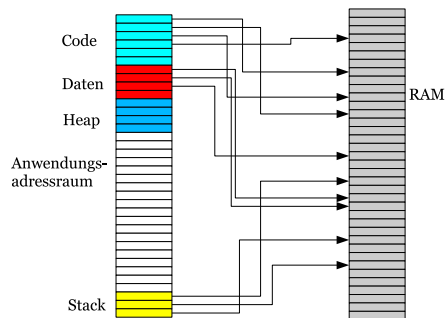
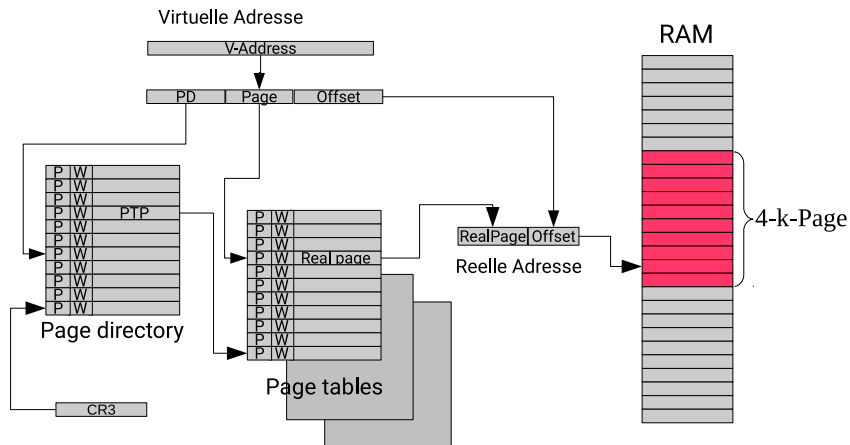


Abbildung des Virtuellen Adressraums

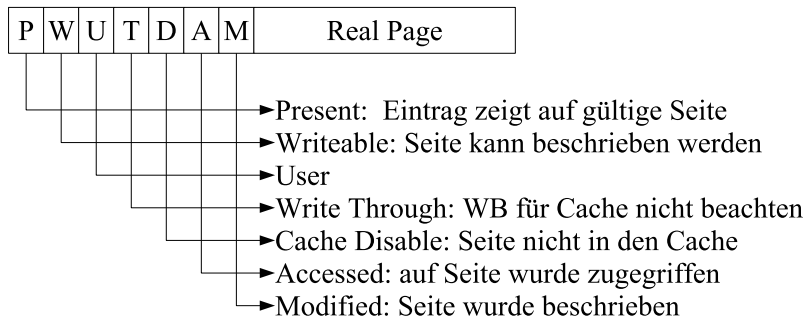
- Jede Seite des Anwendungsadressraums wird auf eine Seite des realen Adressraums abgebildet.
- Dabei muss die Reihenfolge der Adressen der Seiten nicht gleich sein.
- Dies ermöglicht eine sehr platzsparende Speicherverwaltung, maximal eine Seitengröße „Verschnitt“.



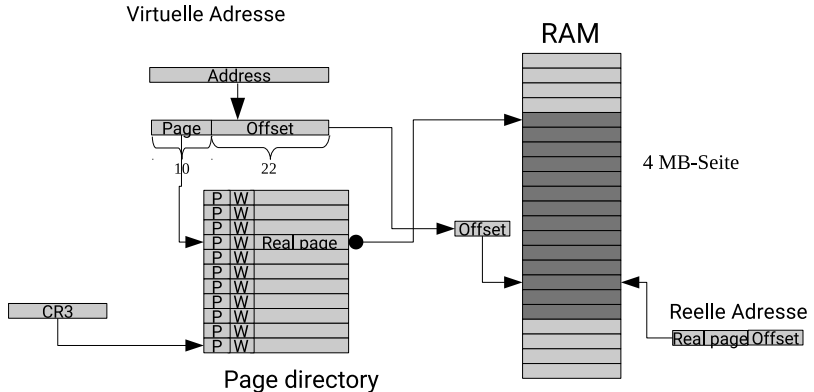
2-stufige virtuelle Adressierung



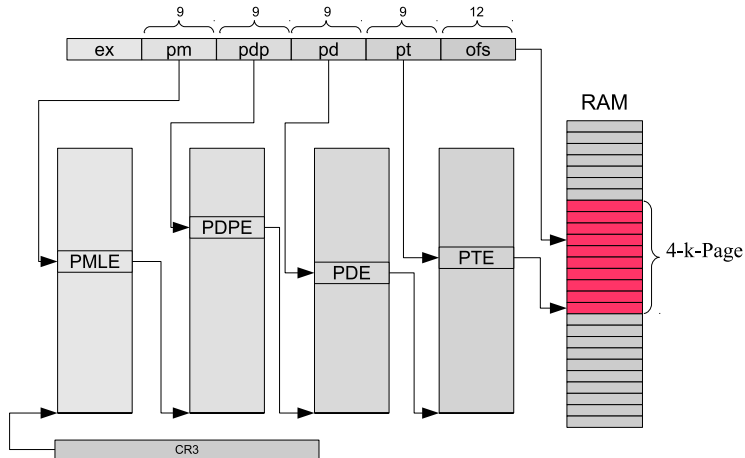
Page Table Entry



Adressierung mit 4MB Pages

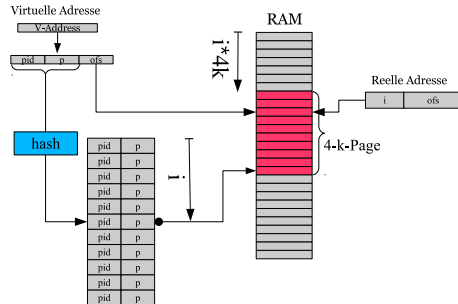


virtuelle Adressierung bei 64-Bit Prozessoren



invertierte Seitentabellen

- virtuelle Adresse wird aufgeteilt, linker Teil wird über Hash-Algorithmus mit Seitentabelle verglichen
- Index des Fundorts in der Tabelle ist die Seitennummer
- Seitennummer wird mit Offset (ofs) zu reeller Adresse ergänzt



Translation Lookaside Buffer (TLB)

- Jede Adressberechnung bedingt einen weiteren Zugriff im Hauptspeicher
- bei mehrstufiger Adressierung entsprechend mehr
- Performanceverschlechterung nicht tragbar
- vollassoziativer Cache für die Zuordnung von virtuellen zu realen Adressen
- Adressberechnung ohne Verzögerung
- „V“ = Gültiger Eintrag

V	VPage	RPage
V	VPage	RPage
V	VPage	RPage
V	VPage	RPage
V	VPage	RPage
V	VPage	RPage