

## E 4 Virtueller Speicher I

### Lernziele

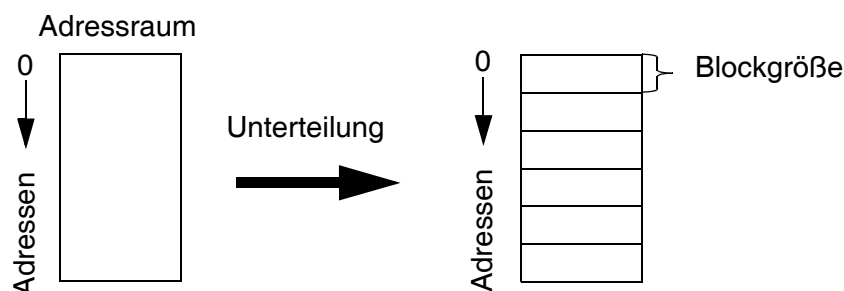
- Funktionsprinzip der MMU zur Blocknummernbestimmung aus der Zugriffsadresse kennen lernen
- Verfahren zur Adressumsetzung anhand praktischer Zahlenbeispiele in ihrer Funktionsweise rekapitulieren
- Sich einen Eindruck über den Platzbedarf für die Verwaltungstabellen abhängig vom Umsetzungsverfahren verschaffen
- Ein- und zweistufige Seitenumsetzungen durchführen
- Die Zusammenhänge in einfachen Darstellungen wiedergeben

### Aufgaben

#### E 4.1 Blockweise Speicherverwaltung

Zur Verwaltung des Speichers ist es sinnvoll, den Adressraum in Blöcke einheitlicher Größe zu unterteilen. Dazu wird eine bestimmte Blockgröße gewählt und der Adressraum beginnend bei Adresse 0 lückenlos mit Blöcken aufgefüllt. Eine MMU (Memory Management Unit) benutzt dieses Verfahren, um eine Adressumsetzung zu realisieren. Dazu muss sie für jede Zugriffsadresse die zugehörige Blocknummer bestimmen können. Um diese Blocknummernbestimmung geht es in dieser Aufgabe.

NB: Solche einheitlich großen Blöcke im Adressraum werden als Seiten (pages) bezeichnet.



- a) Die Blocknummer lässt sich wie folgt aus der Adresse bestimmen:

$$\text{Blocknummer} = \text{Adresse} \text{ DIV } \text{Blockgröße}$$

(DIV: steht für eine Ganzzahldivision ohne Rest)

Welches ist die Blocknummer des Blocks auf der Adresse 0? Blocknr. = .....

- b) Wird als Blockgröße eine Zweierpotenz  $2^n$  gewählt, so lässt sich die Ganzzahldivision zur Bestimmung der Blocknummer besonders einfach ausführen. Wie geht das? (Beschreibung des Verfahrens)

.....

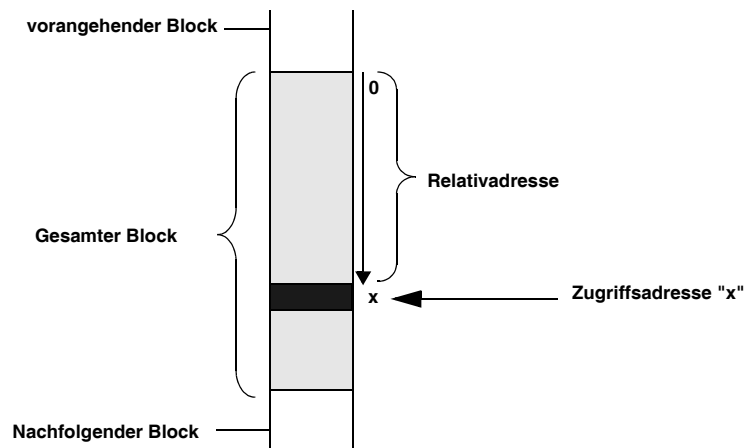
.....

NB: Bei Binärzahlen entspricht 1 gespeichertes Bit genau einer Binärziffer.

- c) Benutzen Sie das Verfahren aus b) um die zur Adresse 0x34af6784 gehörige Blocknummer zu bestimmen, wenn die Blockgröße  $2^{12}$ , d.h. 4 KB, beträgt.

Blocknummer = .....

- d) Wie kann für eine bestimmte Zugriffsadresse die Relativadresse innerhalb des zugehörigen Blocks bestimmt werden? Die Relativadresse ist eine Adresse relativ zum Anfang des Blocks, innerhalb dessen Adressgrenzen die Zugriffsadresse liegt. In untenstehendem Bild ist ein Ausschnitt aus dem Adressraum dargestellt, in dem die Relativadresse zur angenommenen Zugriffsadresse "X" dargestellt ist.



Formel:

.....

- e) Die Blockgröße sei eine Zweierpotenz  $2^n$ . Welches einfache Verfahren kann nun angewendet werden, um die Relativadresse aus der Zugriffsadresse abzuleiten?  
Beschreibung des Verfahrens:

.....

- f) Benutzen Sie das Verfahren aus e) um die zur Adresse 0x34af6784 gehörige Relativadresse zu bestimmen, wenn die Blockgröße  $2^{12}$ , d.h. 4 KB, beträgt.

Relativadresse (Offset) = .....

#### E 4.2 Grundlagen der seitenbasierten Adressumsetzung

Eine bestimmte CPU benutze 32 Bit große virtuelle Adressen und 24 Bit große Speicheradressen. Es komme eine Seitengröße von 2 KB zum Einsatz.

- Wie viele Seiten sind im virtuellen Adressraum vorhanden?
- Wie viele Seitenrahmen sind verfügbar?
- Wie viele Einträge besitzt eine Umsetzungstabelle?
- Wie viele Bits umfasst eine Seitenrahmennummer?
- Wie wird die Relativadresse innerhalb einer Seite umgesetzt? (Als Relativadresse bezeichnen wir die CPU-Adresse abzüglich der Seitenstartadresse.)

## E 4.3: Seitenbasierte Adressumsetzung

Ein Rechnersystem benutze eine seitenbasierte Adressumsetzung mit folgenden Eckwerten:

24 Bit große virtuelle Adressen

24 Bit große Speicheradressen

14 Bit große Relativadressen (d.h. Adressen relativ zum Seitenanfang)

a) Wie groß ist eine Seite?

Seitengröße = ..... Byte

b) Wir betrachten die Ausführung der folgenden Prozessorinstruktion, die bei der Programmausführung auf der Programmadresse **0x02b468** liege: **move.w (a1), d2**

Es handelt sich bei dieser Prozessorinstruktion um einen Datenkopierbefehl folgender Art:

Register **a1** enthält die *Speicheradresse* des Quelloperanden (an dieser Adresse liegt der zu kopierende Datenwert, d.h. es handelt sich um eine Adressierung via einen Zeiger). Das Register **d2** dient als Ziel der Kopieroperation (d.h., dorthin wird der Datenwert kopiert). Bei der Ausführung dieser Prozessorinstruktion sei der Inhalt des Registers **a1: 0x006f0004**

Wie viele Speicherzugriffe sind für die Ausführung dieses Befehls grundsätzlich nötig?

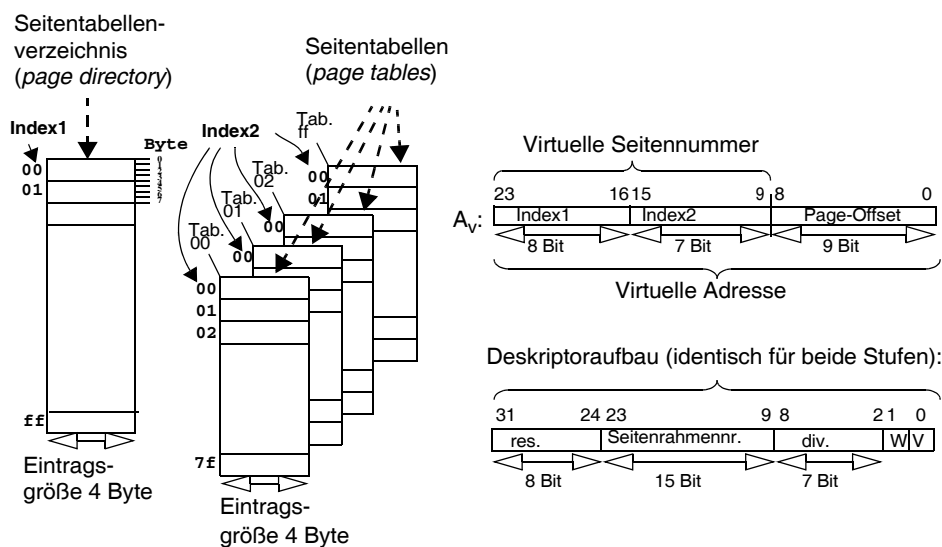
Anzahl Speicherzugriffe: .....

Beim Zugriff auf die virtuelle Adresse **0x006f0004** werde der Seitenrahmen mit der Nummer **0x0b3** benutzt (es handle sich um einen gültigen Zugriff). Auf welcher Speicheradresse erfolgt der Zugriff?

Speicheradresse: ..... (hexadezimal)

## E 4.4 Zweistufige seitenbasierte Adressumsetzung.

Es komme die zweistufige seitenbasierte Adressumsetzung gemäß folgendem Bild zur Anwendung.



Rekapitulation des zweistufigen Adressumsetzungsverfahrens:

Zuerst wird *Index1* der  $A_v$  entnommen und als Index in das *Seitentabellenverzeichnis* benutzt. Aus dem Seitentabellenverzeichnis wird der *Seitendeskriptor* entnommen und aus diesem die *physische Seitennummer* extrahiert. Damit ist der Speicherort der betroffenen Seitentabelle ermittelt (diese beginnt stets am Anfang des referenzierten Seitenrahmens).

Auf diese Seitentabelle wird mit *Index2* aus  $A_v$  zugegriffen und der damit identifizierte Seitendeskriptor entnommen. Dieser dient nun zur eigentlichen Umsetzung der  $A_v$  in eine physische Speicheradresse  $A_p$  ( $A_p$  = Zusammenfügen von Seitenrahmennummer und Relativadresse, d.h. page offset).

NB: Als Tabelleneinträge kommen generell Seitendeskriptoren mit der im Bild dargestellten Struktur zur Anwendung.

- a) Bestimmen Sie den Verwaltungsaufwand (Speicherbedarf für *Seitentabellenverzeichnis* und *Seitentabellen*) für die Adressumsetzung eines Prozesses, der ab Adresse 0 rund 180 KB Code aufweist, ab Adresse 0x800000 rund 400 KB statische Daten speichert und im höchsten Adressbereich rund 100 KB Stack belegt.

Geben Sie den relativen Verwaltungsaufwand als Prozentzahl des Gesamtspeicherbedarfs gemäß folgender Formel an:

$$\text{Relativer Verwaltungsaufwand} * 100\% = V_{\text{erwaltungsaufwand}} / (N_{\text{utzaufwand}} + V_{\text{erwaltungsaufwand}})$$

- b) Skizzieren Sie die benötigten Tabellen, beschriften Sie alle Indexbereiche und geben Sie die Deskriptor-Tabelleninhalte inkl. Schreibschutz- und Gültigkeitsbit soweit bekannt an.

Annahme: Diese zwei Bits belegen die untersten zwei Bitpositionen; die Seitenrahmennummer schließe direkt links daran an, d.h., das Tripel {Seitenrahmennummer, Schreibschutzbit, Gültigkeitsbit} wird als ein einziger zusammengesetzter Binärwert gespeichert.

- c) Wie viel und welcher Teil des virtuellen Adressraumes könnte mit den in a) ermittelten Tabellen umgesetzt werden, wenn man die vorhandenen Tabellen vollständig ausnützen würde? (jedoch keine zusätzlichen Seitentabellen anlegt)?

Angabe der Adressbereiche mit Start-/Endadresse hexadezimal und Größe dezimal in KB.