

Video - RAM

- liegt in jeder Grafikkarte
- damit kann programmiert werden

Ein oder mehrere Bits werden damit für jeden Bildpunkt gesetzt.

Im Textmodus wird der ASCII-Code übertragen damit Buchstaben erzeugt werden. (Pixelmatrix)

Wir nutzen CGA Textmodus

CGA Textmodus

- jede Bildschirmposition 2 Byte im Video-RAM
 - erstes Byte: ger. Offsetadresse, nimmt ASCII-Code auf
 - zweites Byte: unger. Offsetadresse, Vorder-/Hintergrundfarbe

Offset 0: (hex): 68000 adressiert

Ein Byte:

7	6-4	3-0
<u> </u>	<u> </u>	<u> </u>
linker	Hintergrundfarbe	Vordergrundfarbe
	(8 Farben)	(16 Farben)

Cursor

- zum Abfragen der aktuellen Cursor-Position muss Videocontroller programmiert werden.

dafür:

- 1 Datenregister
- 1 Indexregister
- 18 Steuerregister (nur 14, 15 interessant)

Daten; Index register sofort ansprechbar (in/out)
Bei Steuerregister muss dessen Index über Index-
register zuerst angesprochen werden.

1. Was ist ein CGA - Bildschirm
Color Graphics Adapter (erste IBM)

- Standard von 1981
- 16kB Speicher (16384 Byte)
- 640 x 200 Pixel maximum

2. Wie viele Zeichen passen auf einen
CGA-Bildschirm

- 600 x 240 Bildpunkte
- 80 x 25 Zeichen = 2000 Zeichen

3. Wie viel Platz benötigt ein Zeichen?

2 Byte:
(16 Bit)

- jede Bildschirmposition 2 Byte im Video-RAM
 - erstes Byte: ger. Offsetadresse, nimmt ASCII-Code auf
 - zweites Byte: unger. Offsetadresse, Vorder- Hintergrundfarbe

Offset 0: (hex): 08000 adre. sieht

Ein Byte: $\underbrace{7}_{\text{Zeichen}} \quad \underbrace{6-4}_{\text{Hintergrundfarbe (8 Farben)}} \quad \underbrace{3-0}_{\text{Vordergrundfarbe (16 Farben)}}$

4. Wie kann man CGA-Bildschirm ansprechen? Wir nutzen Memory Mapped I/O (über Tiger benutzbar)

```
char* cga_start = (char*) 0x8000;  
cga_start[0] = 'a';  
cga_start[1] = 0b00000100;
```

5. Ablauf des Schreibens eines Zeichens

1. Byte	2 Byte
$[x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8]$	$[x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8]$
$\underbrace{\hspace{10em}}$	
ASCII-Code für bestimmtes Zeichen	x_1 : blinken: 0/1 x_2 : Hintergrundfarbe (8) x_3 : Vordergrundfarbe (16)

6. Ablauf des Setens des Cursors

in-out-Zugriff:

Datenregister (lesen und schreiben, port 3d5)
benötigt um auf adressierte Register zuzugreifen

Indexregister (nur schreiben, port 3d4)

- auswählen des Index des Steuerregisters

- Steuerregister: 14, 15 Byte (16 Bit) für Position des Cursors

Beim Bewegen: Index wird abgegriffen

=> Datenregister gelesen, geschrieben

=> Steuerregister, Position geschrieben

7. Was ist eine Initialisierungsliste in C++?

- kommt nur in Konstruktoren vor

- hat die Aufgabe, die Konstruktion von
Vorfahrenklassen, eingebetteten Elementen/Referenzen
zu regeln

- versorgt also Elemente in einer Liste mit Werten