

VGA – Video Graphic Array

Teil 1: Hardware, Grundlagen und Arbeitsweise

Graphische Betriebssysteme, bunte, unter Umständen bewegte, Graphiken, Bilder in hoher Auflösung, einfache Graphikprogrammierung mittels der OpenGL Modulbibliothek, das ist heutzutage alles eine Selbstverständlichkeit, aber wäre das alles überhaupt entstanden ohne einen firmenübergreifenden Graphikstandard? Betrachten wir zunächst einmal die Entwicklung und die Unterschiede der gebräuchlichsten Standards:

Standard	Auflösung		Farbanzahl		Textmodus	Anmerkung
	horizontal	vertikal	minimal	maximal		
MDA	720	350	Monochrom	Monochrom	nur Text	
Herkules	720	348	Monochrom	Monochrom	Graphik / Text	
CGA	640	200	2	4	Graphik / Text	
EGA	640	350	16	16	Graphik / Text	
VGA	640	480	16	256	Graphik / Text	
SVGA	800	600	16	16	Graphik / Text	Vesa Standard
XGA	1024	768	256	256	Graphik / Text	

Die Besonderheit des VGA Modus ist das er alle vorherigen Modi mit einschließt, also vollständige Kompatibilität zu EGA, CGA, Herkules und MDA gewährleistet. Betrachten wir nur kurz die Rahmenbedingungen die zu der Zeit herrschten:

1. Der zur Verfügung stehende Adressraum war auf 1Megabyte begrenzt, denn es standen ja nur 20 (21) Adressleitungen zur Verfügung. Daher sollte der Speicherbereich der Graphikkarte im Adressraum nur 64kB beanspruchen.
2. Speicher war teuer, deshalb war auch der Graphikspeicher anfangs recht klein.
3. Der Graphikstandard sollte zu den damals üblichen TV Standards kompatibel sein, damit auch eine Ausgabe auf einem Fernsehgerät möglich ist.

Alle diese Einschränkungen setzten dem möglichen Standard eine obere Grenze, die sich natürlich im weiteren Verlauf nach oben verschob und damit neue Standards (SVGA und XGA, usw.) ermöglichte. Und um diese ganzen historischen Betrachtungen einmal abzuschliessen: mit dem Siegeszug graphischer Betriebssysteme wurde ein verbindlicher Graphikstandard endgültig erforderlich.

Graphik Modi:

Da Abwärtskompatibilität ein Entwicklungskriterium war stellt eine VGA Karte verschiedene Videomodi zur Verfügung, z.B.:

Modus 03h: 80 x 25 Zeichen Textmodus mit 16 Farben

Modus 12h: 640 x 480 Pixel Graphikmodus mit 16 Farben

Modus 13h: 320 x 200 Pixel Graphikmodus mit 256 Farben

letzterer ist beispielsweise ein in der Spieleprogrammierung recht beliebter Modus. Der Modus 03h ist zwar ein reiner Textmodus, besitzt aber eine Auflösung von 720 x 200 Pixeln, und stellt für einfachere Graphiken entsprechende Graphikzeichen zur Verfügung. Bei den Betrachtungen der Hardware weiter unten, und den Überlegungen zur Arbeitsweise des VGA Adapters wollen wir allerdings den Modus 12h annehmen.

Sad but true:

Höhere Auflösungen als die oben aufgeführten sind in der Windows Konsole in der Regel nicht benutzbar, da diese selbstverständlich herstellerspezifische Treiber erfordern.

SVGA und XGA sind normalerweise als VESA Modi verfügbar.

Es steckt noch eine Tücke im Detail: Ein Programm in der Windows Konsole das die VGA Graphik verwendet, läuft nur im Vollbildmodus und setzt bedingt das Prinzip des Speicherschutzes unter Windows außer Kraft, da z.B. die meisten Spiele unmittelbare Zugriffe auf das Videoram tätigen (was auch den schnellsten Zugriff ermöglicht).

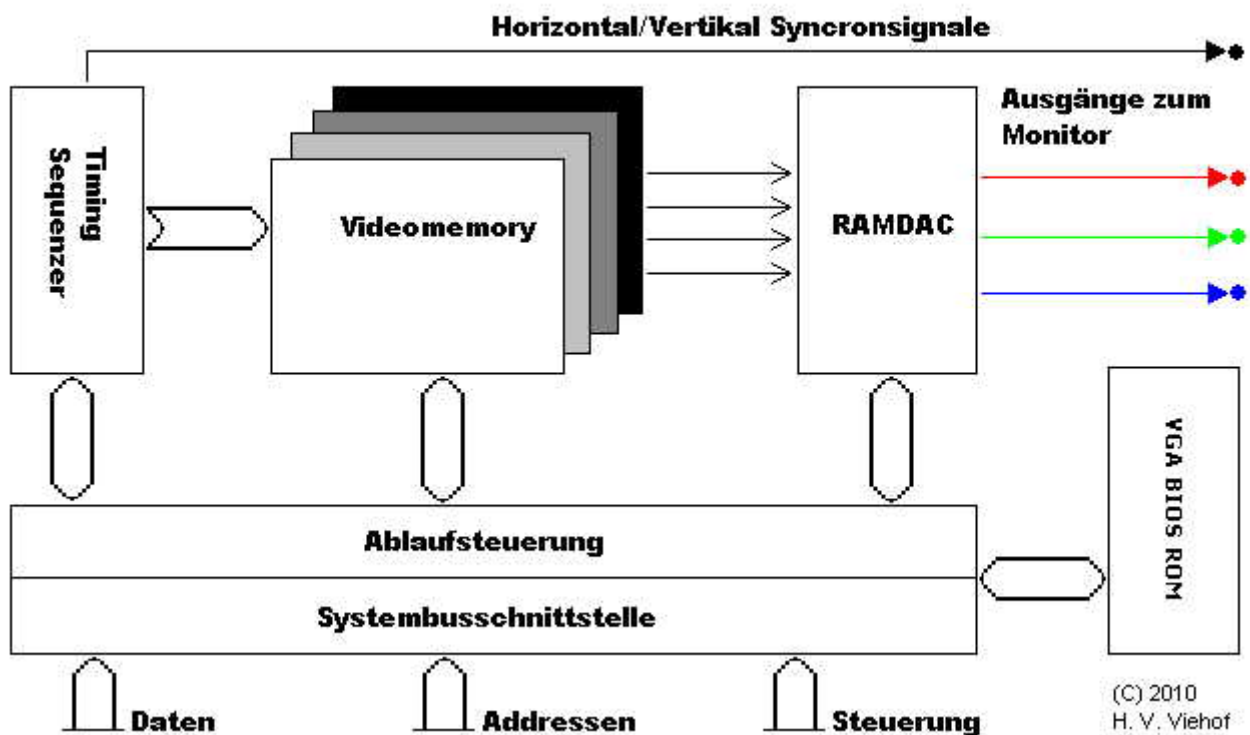
Aufbau der VGA Karte

Wenden wir uns ersteinmal der Hardwareseite zu, aber immer schön der Reihe nach, wir gehen im folgenden nur einmal von den verschiedenen Graphikmodi aus, die verschiedenen Textmodi lassen wir außen vor.

Der Graphikspeicher ('Abbild des Videomemory im Adressraum des PCs) belegt den Adressbereich 0A000h bis 0AFFFh, die Textmodi erfordert drei weitere Funktionsgruppen (in diesem Diagramm nicht enthalten): Zeichentabellen, Zeichengenerator und Textspeicherbereich (0B000h-0B7FFh Text und 0B800 bis 0BFFF Attribute).

Betrachten wir zunächst einmal das stark simplifizierte Schema der VGA Karte:

VGA Adapter Schema



Betrachten wir kurz die einzelnen Baugruppen:

Systembusschnittstelle – hier kommuniziert das System mit den 'Innereien' der Karte, für einen Hochspracheprogrammierer gänzlich uninteressant und für die Lowlevelprogrammierung ist hier nur nützlich zu wissen das man mittels dieser auf die einzelnen Register der VGA Karte mittels IN und OUT Anweisungen, und auf Videoram/Textspeicher mittels MOV Anweisung Zugriff hat.

VGA BIOS ROM – dieses stellt dem System und den Anwendungen Interruptroutinen zur Verfügung die im Prinzip relativ einfache Nutzung der Funktionen ermöglicht, allerdings auf Kosten der Geschwindigkeit.

Ablaufsteuerung – Initialisiert die Karte und steuert den Funktionsablauf, wir wollen diese auf keinen Fall mit einer GPU einer modernen Graphikkarte vergleichen, statt dessen stellen wir uns quasi eine 'Blackbox' vor die hauptsächlich interne Funktionen erfüllt.

Timing Sequenzer – dieser adressiert zyklisch das Videomemory und zwar genau in dem Takt in dem Bildzeilen erzeugt werden sollen. Das bedingt außerdem das der Timing Sequenzer auch den Zeitpunkt von Schreib-/Lesezugriff auf das Videomemory bestimmt. Weiterhin erzeugt er die Zeilen/Bildsynchrone Signale, die der Monitor zum Bildaufbau benötigt.

Videomemory – das ist das Herz (eher das 'Gedächtnis') der VGA Karte.

Dieser Speicher belegt den Adressbereich 0A000h bis 0AFFFh im Adressraum des PCs.

Allerdings sind es genau wie das Schema suggeriert vier Ebenen (Planes) die bildlich gesprochen 'übereinander liegen', und ergeben zusammen einen Speicher von 4 x 64kByte das sind dann 256kByte. Vom Systembus aus kann ich jeweils nur auf eine Ebenen zugreifen, und welche wird durch ein spezielles Register bestimmt. Der Timing Sequenzer adressiert alle Ebenen im Gleichtakt so das am Ausgang vier Bit aus vier 'übereinander liegenden' Speicherstellen geliefert werden, diese verarbeitet dann das:

RAMDAC – übersetzt ungefähr: freiprogrammierbarer Digital Analog Wandler.

Hier wird der vier Bit Wert aus dem Videoram in die Farbe für den Bildpunkt umgesetzt, dazu verwendet der RAMDAC eine Tabelle bzw. Palette.

Diese Palette besteht aus 256 Einträgen, von denen je nach Betriebsmodus 16 oder mehr benutzt werden. Im Graphikmodus 013h (320 x 200 Pixel) werden alle 256 benutzt und im Modus 012h (640 x 480 Pixel) werden die ersten 16 benutzt. Jeder Eintrag besteht aus drei sechs Bit Werten die die Rot, Grün und Blau repräsentieren. Nach der Initialisierung der Karte wird die Palette mit Farben gefüllt, die man dann allerdings durch eine eigene Palette ersetzen kann.

Arbeitsweise der VGA Karte

Fassen wir zunächst einmal das wesentliche zusammen:

Bildpunkte befinden sich in vier zusammengehörenden Bits des Videorams, das in vier Ebenen organisiert ist, diese Speicherstellen des Videorams werden zyklisch vom Timingsequenzer adressiert, und liefern einen vier Bit breiten Datenstream an den RAMDAC. Dieser benutzt die vier Bit (16 Werte) als Zeiger auf die Farbwerte einer Palette. Diese Farbwerte bilden dann drei Analogsignale die auf dem Monitor einen farbigen Bildpunkt erzeugen.

Konkretisieren wir das einmal:

Gehen wir davon aus das wir aktuell Ebene 1 des Videorams beschreiben können, und setzen einen Bitwert von 10000000xb in das erste Byte des Videorams, also nach 0A000h, das würde jetzt einen Bildpunkt in der linken oberen Ecke des Bildschirms erzeugen. Soweit, so gut, aber: welche Farbe hat der Punkt nun? Wir hatten vorausgesetzt daß Ebene 1 beschrieben wird, und falls Ebenen 2, 3 und 4 keine Punkte enthalten, also eine Null liefern, ist der vier Bit Wert der ans RAMDAC geliefert wird: 0001xb, damit würde Farbwert 1 der Palette gefordert, und sagen wir einmal das wäre zu diesem Zeitpunkt folgendes Wertetrippl: 0, 0, 63 dann wäre der Punkt ein sattes Blau.