

Aufgabe 4: Bildschirmansteuerung

G. Kemnitz, C. Giesemann, TU Clausthal, Institut für Informatik

13. Dezember 2017

Zusammenfassung

Vorgegeben ist eine einfache Bildschirmansteuerung (VGA-Modus), mit der beispielhaft Bildelemente dargestellt werden. Zu entwerfen ist eine einfache Animation, bei der ein kleines Bild mit Hilfe von Tasten auf dem Bildschirm bewegt werden kann.

Funktion

Auf der Versuchsbaugruppe befindet sich auch ein VGA-Anschluss zum Anschluss eines Computerbildschirms. Die Bildschirmschnittstelle ist für klassische Monitore mit Bildröhre ausgelegt. Das VGA-Signal kann aber auch an den VGA-Eingang eines TFT-Monitors angelegt werden.

In einer Bildröhre wird das Bild zeilenweise mit einem Elektronenstrahl geschrieben. An den 3 Kathoden (je eine für die Grundfarben rot, grün und blau) werden je nach Bildpunkthelligkeit und -farbe in über das rgb-Signal einstellbarer Menge Elektronen abgestrahlt und durch die Anodenspannung von etwa 20.000 V in Richtung Bildfläche beschleunigt. Auf dem Weg zur Bildfläche wird der Elektronenstrahl durch Elektromagnete abgelenkt. An der Aufschlagstelle der Elektronen auf der Bildfläche entsteht ein leuchtender Punkt (Abb. 1).

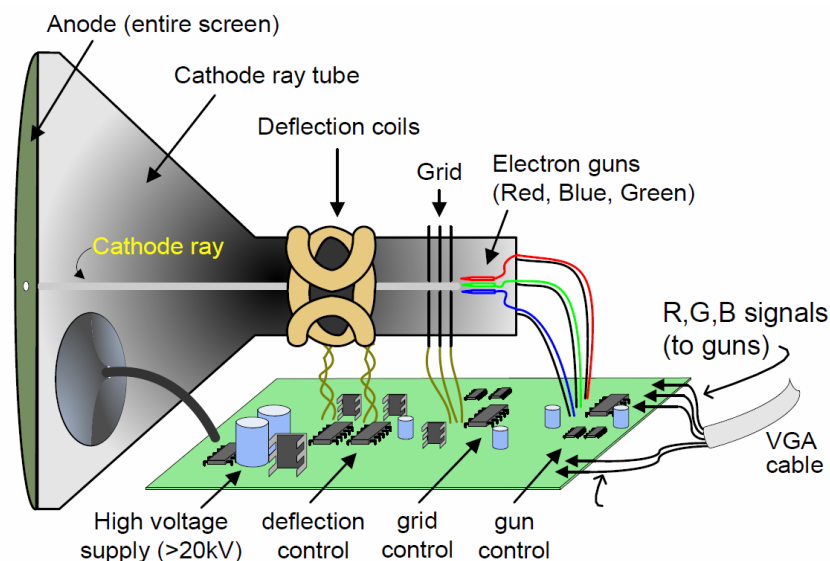


Abbildung 1: Funktion eines veralteten Röhrenmonitors [Nexys3 RM, S.15]

Im VGA-Modus besteht ein Bild aus 480 Zeilen zu je 640 Bildpunkten, die der Strahl nacheinander schreibt. Zwischen den Zeilen kehrt der Strahl zum Zeilenanfang zurück. Während des Strahlrücklaufs ist der Strahl unsichtbar, d.h. der Kathodenstrom ist gleich Null (Abb. 2).

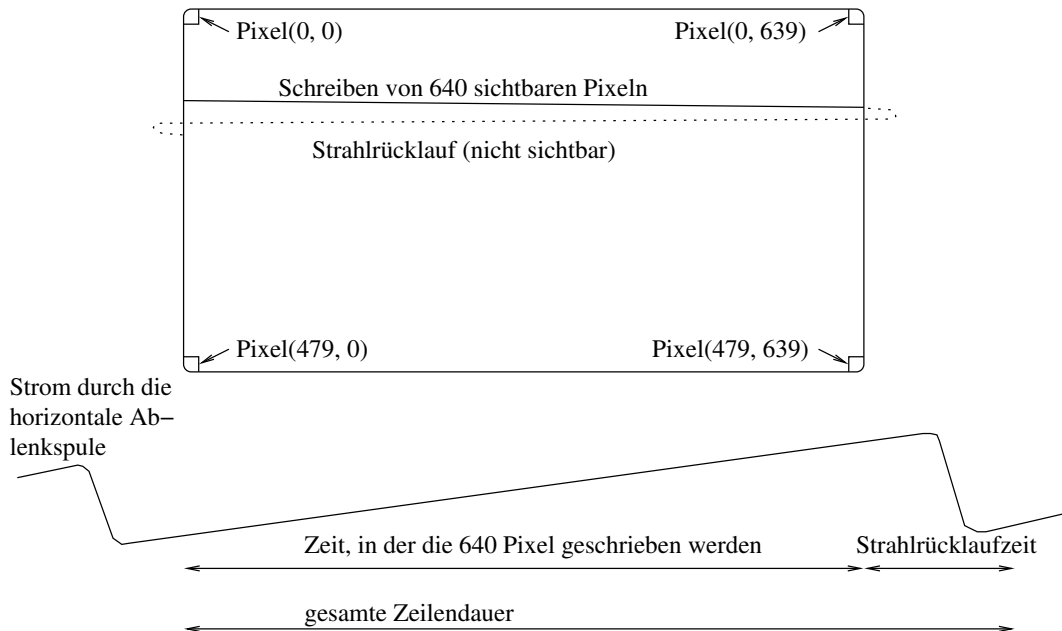


Abbildung 2: Zeilenweises Schreiben der Bildpunkte

Die Grafikkarte sendet an den Monitor (vgl. Abb. 3):

- einen kurzen Sync.-Impuls zum Zeitpunkt des Strahlrücklaufs
- das rgb-Signal in Form von 3 Spannungswerten im Bereich von 0 bis 1V (während des Strahlrücklaufs 0V).

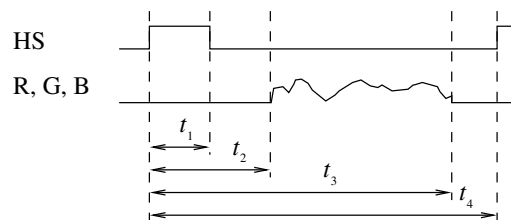


Abbildung 3: VGA-Steuersignale für eine Zeile

Für eine Auflösung 640×480 und 60 Bilder pro Sekunde gelten folgende Zeiten:

	Dauer	Takte ($f_{Pixel} = 25 \text{ MHz}$)
t_1	$3,84 \mu\text{s}$	96
t_2	$5,76 \mu\text{s}$	144
t_3	$31,36 \mu\text{s}$	784
t_4	$32 \mu\text{s}$	800

Die Vertikalansteuerung funktioniert ähnlich. Zu Beginn eines jeden Bildes ist ein vertikaler Sync.-Impuls zu erzeugen, gefolgt von wenigen dunkel getasteten Zeilen. Darauf folgen 480 sichtbare und einige wenige unsichtbare Bildzeilen (Abb. 4).

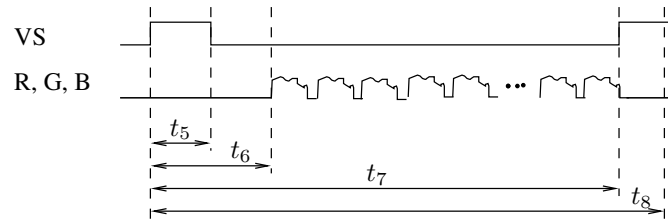


Abbildung 4: VGA-Steuersignale für ein Bild

	Dauer	Zeilen ($f_{Zeile} = \frac{1}{32\mu s} = 31,25\text{kHz}$)
t_5	64 μs	2
t_6	992 μs	31
t_7	16,352ms	511
t_8	16,7ms	521

Die RGB-Signale werden über das Widerstandsnetzwerk in Abb. 5 erzeugt. Rechts in der Abbildung ist die Steckerbelegung des VGA-Anschlusses gezeigt.

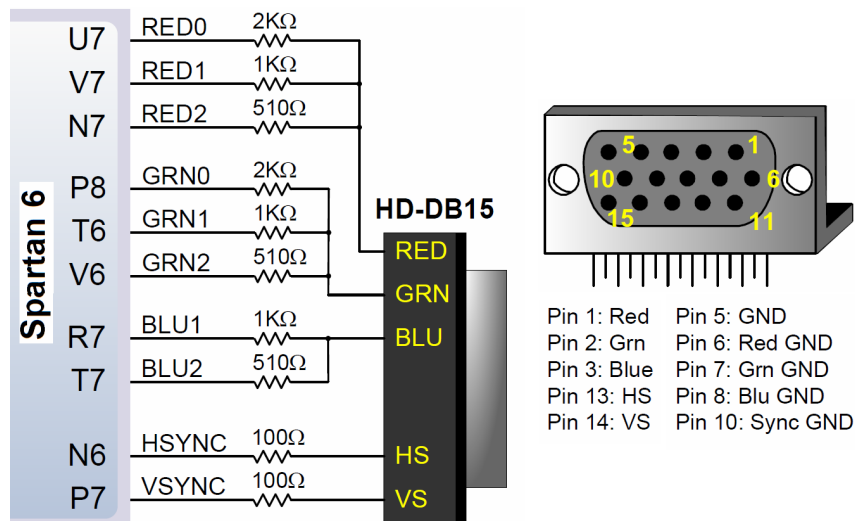


Abbildung 5: VGA-Steuersignale für ein Bild

Eine VGA-Ansteuerschaltung benötigt mindestens einen Takteingang, 2 Sync.-Ausgänge und Ausgangssignale für die Helligkeit und Farbe der zu schreibenden Bildpunkte:

```

1 entity vga is
2   port(
3     clk : in std_logic;    -- 100MHz Baugruppentakt
4     hSync : out std_logic; -- horizontaler Sync-Impuls (Zeile)
5     vSync : out std_logic; -- vertikaler Sync-Impuls (Bild)
6     rgb : out std_logic_vector(7 downto 0) -- RGB-Signal
7   );
8 end entity;
```

Die Farbwerte sind als Konstanten vereinbart. Jeder Farbwert setzt sich aus drei Bit für rot, drei Bit für grün und zwei Bit für blau zusammen. Für die Grundfarben gilt:

```

1  — Grundfarben
   "RRRGGGBB"
2  constant schwarz : std_logic_vector(2 downto 0) := "00000000";
3  constant rot : std_logic_vector(2 downto 0) := "11100000";
4  constant gruen : std_logic_vector(2 downto 0) := "00011100";
5  constant blau : std_logic_vector(2 downto 0) := "00000011";
6  constant gelb : std_logic_vector(2 downto 0) := "11111100";
7  constant cyan : std_logic_vector(2 downto 0) := "00011111";
8  constant violett : std_logic_vector(2 downto 0) := "11100011";
9  constant weis : std_logic_vector(2 downto 0) := "11111111";

```

Die Ansteuerschaltung selbst benötigt einen Spaltenzähler und einen Zeilenzähler. Der Spaltenzähler zählt mit dem 25MHz-Pixeltakt. Der Zeilen-Sync.-Impuls wird mit Rücksetzen des Spaltenzählers eingeschaltet und bei Zählerstand 95 ausgeschaltet. Das Signal für die Dunkeltastung während des Zeilenrücklaufs wird bei Zählerstand 143 auf Null und bei Zählerstand 783 auf Eins gesetzt. Bei Zählerstand 799 wird der Spaltenzähler rückgesetzt. Der Zeilenzähler schaltet bei jedem Zeilenwechsel weiter und steuert in ähnlicher Weise die zugehörigen Sync.- und Dunkeltastsignale.

Bildelemente dürfen nur im Bereich sichtbarer Pixel erzeugt werden, d.h. nur wenn die beiden Dunkeltastsignale Null sind. Zur Erzeugung von Linien werden Zeilen- und Spaltenzähler mit vorgegebenen Werten verglichen. Ein gespeichertes Bild wird angezeigt, indem ein Speicher mit der Spalten- und Zeilennummer adressiert und der adressierte Speicherinhalt als Bildsignal ausgegeben wird. Die Datei vga.vhd aus der zip-Datei zeigt das am Beispiel.

Aufgabe 4.1: Anlysieren der vorgegebenen Schaltung vga.vhd

1. Was für Bildelemente werden von der Schaltung vga.vhd erzeugt?
2. Wozu dient die Anweisung “vsl_vCt := to_unsigned(vCounter, 10);”?
3. Warum wird das Bild in dem kleinen Speicher gespiegelt? Wird es horizontal, vertikal oder an der Diagonale gespiegelt?
4. Wie groß ist das gespeicherte Bild (Zeilen- und Spaltenzahl) und wie groß wird es dargestellt?

Aufgabe 4.2: Ausprobieren der vorgegebenen Schaltung vga.vhd

Erstellen Sie mit den Dateien vga.ucf und vga.vhd aus der zip-Datei auf der Web-Seite ein neues ISE-Projekt. Übersetzen Sie es und programmieren Sie den Schaltkreis. Zum Testen ist an den VGA-Anschluss ein Monitor anzuschließen (17" TFT aus dem Nachbarlabor auf dem Schrank link).

Aufgabe 4.3: Erweitern der Beispielschaltung

1. Ändern Sie das Beispielprogramm so, dass nur noch ein schwarzes Raster auf weißem Hintergrund dargestellt wird. Abstand der Rasterlinien 10 Pixel.
2. Fügen Sie zu dem Raster ein Cursorbild in der Bildschirmmitte hinzu. Das Bild, z.B. ein Kreis, soll als Bitmap wie im Beispielentwurf im Festwertspeicher stehen.
3. Als nächstes sollen sich die Cursorkoordinaten über Taster mit einer Geschwindigkeit von etwa 4 Pixeln pro Sekunde ändern lassen:
 - BTNR: nach rechts

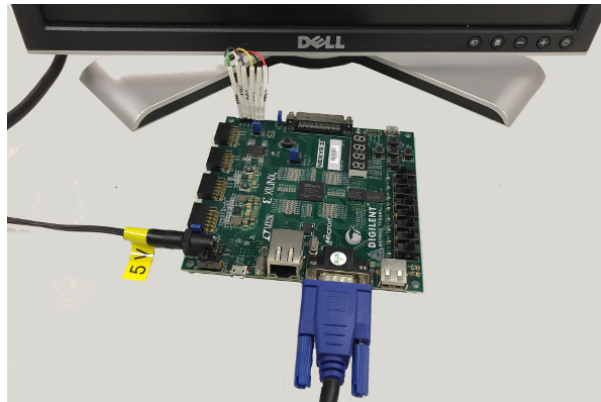


Abbildung 6: Anschluss der Tastatur und des Logikanalysators

- BTNL: nach links
- BTND: nach unten
- BTNU: nach oben

Der Cursor soll sich nicht aus dem Bild bewegen.

Abgabekriterien

- funktionierende Bildschirmdarstellung