

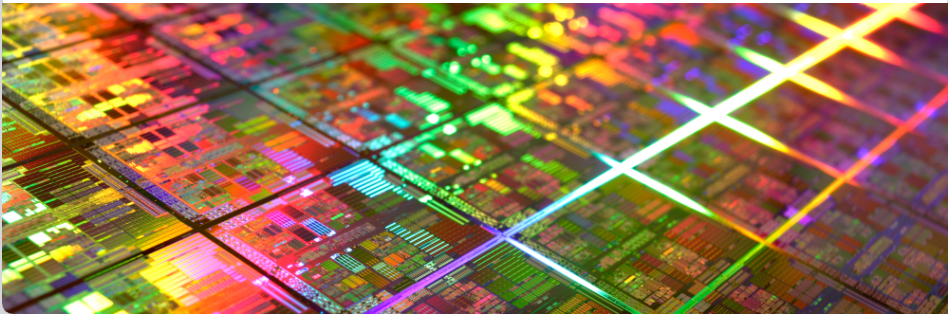
Basispraktikum Technische Informatik

VGA-Ansteuerung

Michael Bromberger

Lehrstuhl für Rechnerarchitektur und Parallelverarbeitung

3. Juni 2015



Anmeldung

- Es fehlen noch 4 Anmeldungen
- Bitte schaut nach, ob Ihr euch noch anmelden müsst

VGA-Ansteuerung

- Entwurf einer Schaltung
- Ansteuerung eines VGA-Monitors
- ⇒ VGA-Signalgenerierung
 - Ausgabe geometrischer Körper
 - Beeinflussung deren Lage über Nutzereingaben

Bilderzeugung

- ① VGA-Signale generieren
 - Farbwerte aller Bildpunkte (Pixel) direkt setzen
 - Signale für die zeitliche Synchronisation

- ② Sprites
 - in Hardware vordefinierte Pixel platzieren
 - ⇒ direkt in die VGA-Signalgenerierung einblenden/-mischen

- ③ Bildspeicher
 - Vorausberechnung
 - Gesamtes Bild in Bildspeicher speichern
 - Ganzes Bild auslesen
 - ⇒ VGA-Signalgenerierung

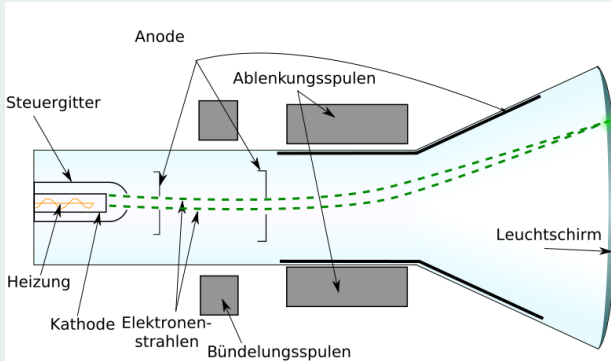
Bildspeicher

- Teil des Video-RAM von Computern
- entspricht einer digitalen Kopie des Monitorbildes
- jedem Pixel des Bildschirms ist ein bestimmter Bereich des Speichers zugewiesen
- Größe des Speichers
 - ⇒ Auflösung
 - ⇒ Farbtiefe

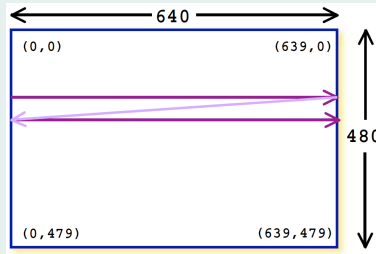
Sprites

- Vordefinierte Pixel, die in/über die sonstige Anzeige gelegt werden
- Keine dynamische Berechnung
- Der Name kommt daher, dass ein Sprite im Grafikspeicher nicht zu finden ist, sondern auf dem Bildschirm „umherspukt“
- Die Platzierung bzw. das Verschieben erfolgt durch die Grafikhardware
- Hardware-Sprites werden erst zum Anzeigzeitpunkt in den Datenstrom eingeblendet
- Mehrere Sprites können sich überlagern

Aufbau einer Bildröhre



Aufbau eines sichtbaren Bildes



- zeilenweise Bewegung von links nach rechts
- anschließend Rücklauf und Anfang einer neuen Zeile
- 640 sichtbare Pixel pro Zeile, 480 Zeilen, danach von vorn
- 60 Bilder pro Sekunde (~ 60 Hz) \Leftrightarrow 25,175 MHz Takt (~ 40 ns)
- Zeitdauer für den gesamten Bildaufbau ca. 17 ms

Signale

Beim Ansteuern eines VGA-Ausgangs müssen 5 Signale generiert werden.

- **hsync**: horizontal sync
- **vsync**: vertical sync
- **R**: rot
- **G**: grün
- **B**: blau

RGB-Signale

- Die RGB-Signale sind analoge Signale
 - Ihr Wert bestimmt die Stärke des Elektronenstrahls und damit die Helligkeit.
 - Die Signale können bei unseren FPGA-Karten durch einen 4-Bit-Zahlenwert pro Farbe bestimmt werden
 - 4 Pins am FPGA pro Farbe
- ⇒ pro Farbe 16 mögliche Werte, insgesamt 4096 Farbwerte
- Die Umwandlung in jeweils ein analoges Signal pro Farbe erfolgt mit Hilfe eines Digital-Analog-Umwandlers (DAC) direkt auf der Karte

hsync

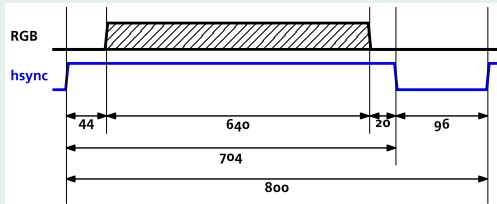
Das Signal **hsync** kennzeichnet das Ende einer Zeile und bewirkt den Rücklauf des Elektronenstrahls und den anschließenden Beginn einer neuen Zeile.

vsync

Nach der Ausgabe der Zeilen wird durch Setzen von **vsync** signalisiert, dass in die linke obere Ecke des Bildes zurückgesprungen werden soll.

- hsync und vsync sind „digitale“ Signale
- im Modus 640×480 sind beide Signale low-aktiv, d.h. der Puls erfolgt durch Setzen einer '0', ansonsten ist das Signal auf '1' gesetzt.

Zeitlicher Ablauf – hsync

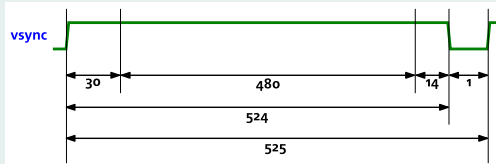


■ Taktfrequenz: 25,175 MHz

- Bildinformation wird nur während der 640 Takte gesetzt
- Die restlichen Takte sind die RGB-Signale auf schwarz gesetzt
- Die '0' von hsync ist genau 96 Takte lang
- Puffer vor und nach dem sichtbaren Bereich

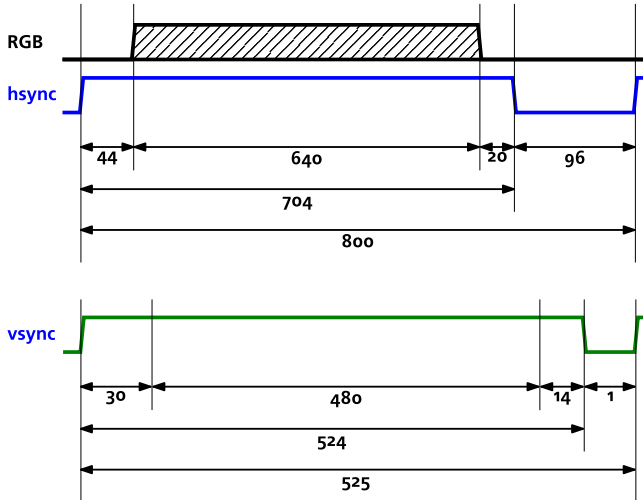
⇒ pro Zeile 800 Takte – nicht nur 640!

Zeitlicher Ablauf – vsync

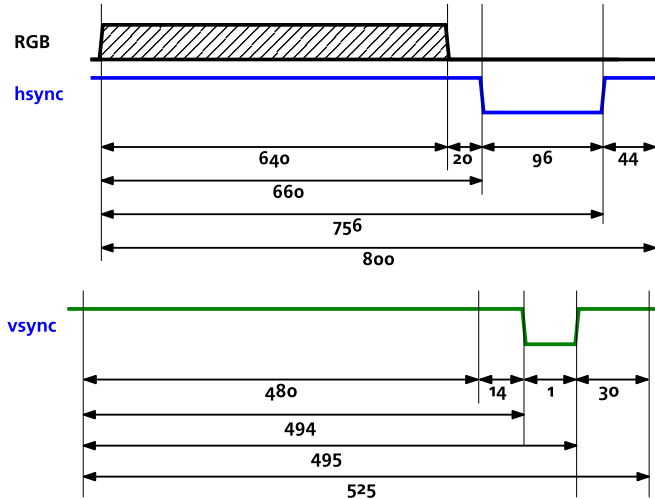


- **Zahlen beziehen sich auf Zeilen**, nicht auf die 25,175 MHz
 - hsync als Takt für vsync? \Leftrightarrow 25,175 MHz-Takt + Zeilennummer
 - während der 480 Zeiteinheiten werden die Zeilen dargestellt
 - vsync wird hier eine Einheit lang gesetzt
- ⇒ Dauer: $1 * 800$ Takte des 25,175 MHz-Taktgebers

VGA-Signal



VGA-Signal – andere Zählweise



Hinweise

- Angaben im vorherigen Schaubild sind Zeitspannen
- Zählung der Takte beachten (ab 0 oder 1, aber einheitlich)
- Generierung von `vsync` hängt mit Signal `hsync` zusammen und nicht vom Ende der 800 Takte einer Zeile.
- `hsync` wird während `vsync` nicht unterdrückt
- Sinnvolle Trennung in verschiedene Prozesse
- Auswahl der richtigen Bedingungen in den Prozessen (auf 0, 1 oder 800 testen?)
- Als Taktgeber den Quarz mit 25,175 MHz und nicht den mit 125 MHz (+ DCM) der FPGA-Karte verwenden!

Simulation

- **Simulation der Teilaufgaben/-schaltungen**
- **Simulation der Gesamtschaltung**
 - Fehler frühzeitig erkennen
 - umständliche Fehlersuche und zeitaufwendige Tests
 - Wartezeit bei der Synthese
 - Meist keine Anzeige des Bildes
- **Simulationsdauer** entsprechend wählen
 - Dauer pro Zeile (hsync): $\sim 32 \mu\text{s} = 32.000 \text{ ns}$
 - Aufbau pro Vollbild (vsync): $\sim 17 \text{ ms} = 17.000.000 \text{ ns}$
- Wenn nötig **Simulation nach der Synthese**
 - Post-Synthesis / Translate / Map / Place & Route - Simulation

VGA-Ansteuerung

- ① VGA-Ausgang
- ② Farbverlauf
- ③ Rechteck
- ④ Beeinflussung der Körper

- Baut aufeinander auf

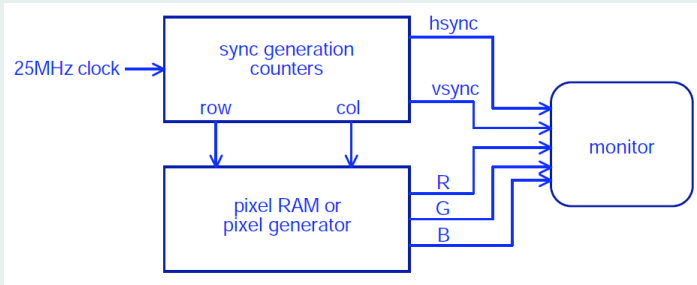
- **Pro Aufgabe je eine extra Schaltung entwerfen!**

⇒ 4 Projekte, 4 Simulationen, 4 Bitstream-Dateien

- VGA-Kabel, Eingang mit +/- umschalten, „Auto“
- 1-Pixel breiter Rahmen auch bei Aufgaben 2-4

Gesamtschaltung

Ein vereinfachtes Schaubild der Gesamtschaltung ohne Steuerung:



Basispraktikum Technische Informatik

VGA-Ansteuerung

Michael Bromberger

Lehrstuhl für Rechnerarchitektur und Parallelverarbeitung

3. Juni 2015

