fbas_enc: Farbvideosignalerzeugung mit einem CPLD

V0.21 (c) 2007 Jörg Wolfram

1 Lizenz

2 Geschichte und Features

Nach einigen Projekten zur Farbausgabe mit AVR-Mikrocontrollern war auch die Frage entstanden, ob man auf einfachem Wege ein FBAS-Signal erzeugen könnte. Zum Beispiel, um ein moduliertes HF-Signal in eine Antennenanlage einspeisen zu können. Da ein Mikrocontroller dafür aufgrund der notwendigen Verarbeitungsgeschwindigkeit eher ungeeignet ist, fiel die Entscheidung zugunsten eines CPLD. Natürlich gibt es für solche Dinge auch Spezial-IC's, aber diesen Weg wollte ich nicht wählen. Herausgekommen ist letztendlich ein sehr simples Design, welches auch in einen XC9536 von Xilinx passt. Der Encoder kann auch in andere Designs integriert werden, vorausgesetzt, die Bestimmungen der LGPL werden eingehalten.

- Gewinnung der Farbträgerfrequenz aus dem Systemtakt
- Eingänge: hsync,vsync und rgb
- 8 Grundfarben / 8 Graustufen wählbar
- Gleiche Hardware kann für PAL und NTSC verwendet werden
- einfache Einbindung als Komponente in andere Designs

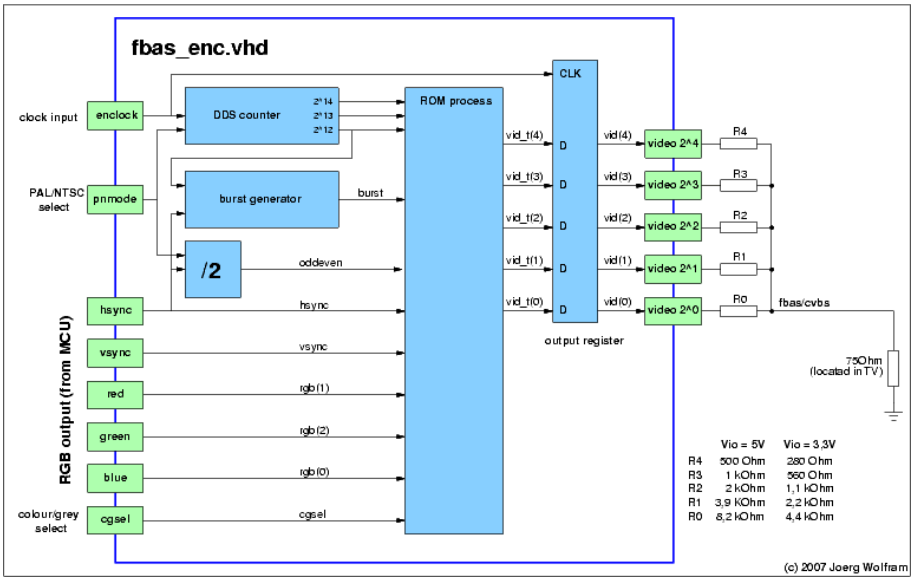
3 Ein bisschen Theorie

Zum Beschreibung von Farbvideosignalen gibt es viele Adressen im Internet und so will ich nur kurz das Wichtigste anhand des PAL-Systems anreissen.
Horizontal- und Vertikaltiming sind dem schwarzweissen BAS-Signal weitestgehend identisch, was hinzukommt, ist der Farbträger. Damit dieser auf SW-Geräten möglichst nicht sichtbar ist, und um Moires zu vermeiden wurde eine „krumme" Frequenz gewählt. Und zwar die Horizontalfrequenz von 15625 Hertz * 283,75 plus die halbe Verikalfrequenz (25 Hertz). Damit kommt man auf 4433618,75 Hertz. Dafür könnte man nun einen Quarzoszillator benutzen oder gewinnt die Frequenz aus dem Systemtakt. Dies geschieht mittels DDS (Direct Digital Synthesis) wobei ein Fehler von eben den obengenannten 25 Hz in Kauf genommen werden muß, oder man benötigt sehr breite Zähler zur Signalerzeugung. Bei einer Taktfrequenz von 16 MHz braucht man so einen Zähler (Akkumulator) mit 12 Bit Breite. Bei jedem Takt wird zum Zähler 1135 (283,75 * 4) dazuaddiert, und schon hat man die (ungefähre) Farbträgerfrequenz. Bei 20 MHz wären es dann theoretisch 908, es würde auch ein 10 Bit breiter Zähler mit Addition von 227 genügen. Das Ganze funktioniert dann so, dass der Zähler zu jedem Abtastzeitpunkt genau den Wert hat, als würde er mit dem 4096-fachen der Farbträgerfrequenz getaktet.
Um ein Farbsignal zu identifizieren und den Quarzoszillator im TV zu synchronisieren, wird der ansonsten quadraturmodulierte Farbträger kurz nach dem Synchronimpuls für ca. 10 Perioden als sog. Burst gesendet, wobei die Phase von Zeile zu Zeile zwischen -135 und +135 Grad wechselt. Nach meinen Erfahrungen reichen aber auch 8 Perioden aus. Die Amplitude des Bursts beträgt mit +-0,15V genau 50% der maximalen Amplitude bei maximaler Farbsättigung. Für die Grundfarben reicht eine feste Sättigung von 50% vollkommen aus, wodurch sich das Ganze noch weiter vereinfacht.
Das FBAS-Signal setzt sich aus dem Helligkeitssignal Y [0,299*R + 0,587*G + 0,114*B] und den beiden Farbartsignalen U [0,493 * (B-Y)] und V [0,877 * (R-Y)] zusammen. Das Y-Signal wird direkt übertragen, mit den U- und V-Signalen wird der Farbträger quadraturmoduliert. Kurzgefasst wird das U-Signal mit dem Farbträger multipliziert und

das V-Signal mit dem um 90 Grad oder 270 Grad phasenverschobenen Farbträger. Dabei wird der Farbton in der Phase und die Farbsättigung in der Amplitude des modulierten Farbträgers kodiert. Der modulierte Farbträger wird dann zum Y-Signal hinzuaddiert und fertig ist das FBAS-Signal. Naja, nicht ganz, während der Synchronimpulse muss es auf 0V gesetzt werden.

4 Die Realisierung

Nun stellt sich die Frage, wie man das alles in ein CPLD hineinbekommt. Der Farbträger und das Burst-Signal machen die wenigsten Probleme, die Modulation und Mischung mit dem Y-Signal ist schon weniger trivial. Bei den ersten Versuchen zeigte sich, dass es theoretisch möglich ist, ein FBAS-Signal zu erzeugen, indem man den Signalpfad digital nachbildet. Allerdings ist der Logikaufwand recht hoch und. Daraus wurde die Idee geboren, die Berechnungen schon vorher durchzuführen und in einem laaangen case-statement als ROM zu verpacken. Als Ausgang wird ein 5 Bit breites Binärsignal geliefert, welches mit dem Systemtakt in das Ausgangsregister übernommen wird. Ein paar Widerstände an den Ausgängen, die sich durch Serien- und Parallelschaltung normaler E12-Widerstände realisieren lassen, erzeugen dann das FBAS-Signal an 75 Ohm Last. Dabei müssen die Ausgänge in der Lage sein, 10mA bei H-Signal zu treiben. Dazu müssen (zumindest beim eingesetzten CPLD) die Ausgänge auf STD-Mode konfiguriert sein, was in der ucf-Datei auch geschehen ist. Die Version für 3,3V IO-Spannung sind nur berechnet, aber nicht getestet.



5 Fazit und Ausblick

Das in VHDL geschriebene Design passt in ein XC9536 CPLD, sofern nur PAL oder nur NTSC erzeugt werden soll. Leider ist auf den manchen TV-Geräten bei machen Farben ein leichtes, schnell wanderndes Muster zu sehen, dessen Ursache ich noch nicht genau bestimmen konnte. Es kann am Verhältnis von Farbträger zur Horizontalfrequenz liegen oder auch am Experimentierboard-Aufbau. Dritte Möglichkeit wären die Filter in den TV-Geräten. Beim Anschluß an einen AVR-Mikrocontroller war der interne Quarzoszillator erst nach „Ziehen" für ein einigermassen stabiles Farbbild zu gebrauchen, ein externer Quarzoszillator (16,20,32 oder 40 MHz) ist auf jeden Fall sinnvoll, wobei das CPLD auch die Funktion des Vorteilers mit übernehmen kann.

6 Changelog

18.2.2007 erste öffentliche Version (0.21)

- 16 und 20 MHz Version