

fbas_enc: FBAS(CVBS) encoder for PAL and/or NTSC

V0.31 (c) 2007 Jörg Wolfram

1 License

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

2 History and features

After some projects of rgb colour video output with AVR microcontrollers there was the question about FBAS (cvbs) signal generation on a simple and easy way. For example, to be able to feed a modulated HF signal into an aerial arrangement.

Because a microcontroller is rather inexpedient for it on account of the necessary processing speed, the decision fell in favour of a CPLD. Of course there is for such things also special-IC's, but I did not want to choose this way. As result there is a simple design which also fits in a XC9536 from Xilinx. The encoder components can be also integrated into other designs, provided, the regulations of the LGPL are kept.

- the colour carrier frequency is generated from the system clock of 16 or 20 MHz
- inputs: hsync, vsync and rgb
- 8 basic colours / 8 grey steps eligible
- the same hardware can be used for PAL and NTSC

3 A little bit of theory

There are many usefull addresses on the Internet and so I only want to tell briefly the most important to know about the PAL system. Horizontal timing and vertical timing are identical to the black and white BAS signal as much as possible. The main difference is the colour carrier signal which is used to encode the additional information. To avoid moires on black'n white pattern the colour carrier has a crooked frequency. Namely the horizontal frequency of 15,625 hertz * 283.75 plus the half vertical frequency (25 hertz). With it one comes on 4,433,618.75 hertz. To generate this we can use a quartz oscillator or find a way to use the system clock from teh rgb source (e.g. microcontroller). This can be done by using DDS (Direct Digitally Synthesis) and a error of just abovementioned 25 hertz must be accepted, otherwise we need very wide counters to the signal production. With a clock frequency of 16 MHz we need such a counter (accumulator register) with 12-bit width. Every clock the counter adds 1135 (283.75 * 4) and so we get the (approximate) value of the colour carrier. With 20 MHz clock speed this would be theoretical 908, a 10-bit-wide counter should be also enough with addition of 227.

To identify a colour signal and to synchronise the quartz oscillator in the TV, the normally quadrature-modulated colour carrier will be sent unmodulated shortly after the synchronous impulse for approx. 10 periods. This is named as the **burst**. If burst is sent the phase changes from line to line between -135 and +135 degrees. However, after my experiences 8 periods are also sufficient. The amplitude of the Bursts amounts to 0,15V exactly 50of the maximum amplitude with maximum colour saturation. For the basic colours a saturation of about 50completely sufficient but its also possible to create a table with another saturation. The FBAS (CVBS) signal consists of three signals:

- the luminance signal $Y = 0,299 * R + 0,587 * G + 0,114 * B$

- the U colour signal $U = 0,493 * (B-Y)$
- the V colour signal $U = 0,877 * (R-Y)$

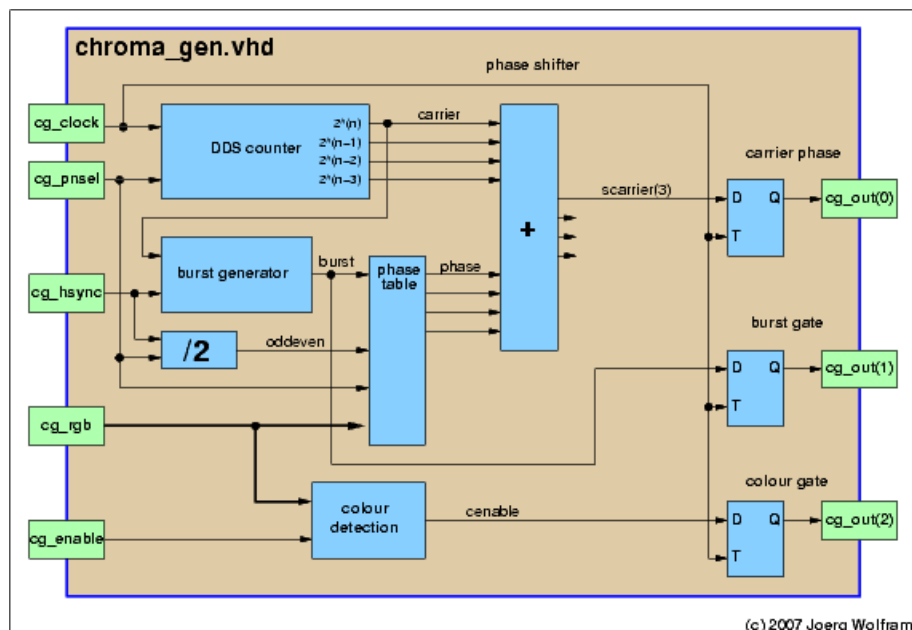
The Y signal will transfer directly, the U and V signals modulates the colour carrier by quadrature-modulation. In brief the U signal is multiplied by the cosinus wave of the colour carrier and the V signal with about 0 or 180 degrees of the sinus wave of the colour carrier. Besides, the tone is encoded in the phase and the colour saturation in the amplitude of the modulated colour carrier. In the last stage the three signals will be added and during the synchronous impulses it must be set to 0 V.

4 Realisation

And now the big question, how can we fit these functions in a small CPLD. The colour carrier and the Burst signal do the least problems, the modulation and mixture with the Y signal is less trivial. After some experiences with long ROM tables including the first official version 0.21 I decided this was not a good way. Poor image quality and the need for an external script were the reasons for searching another way. With the first attempts appeared that it is possible theoretically to generate a FBAS (CVBS) signal, while rebuild the signal path digitally. But, the logic expenditure is rather high. The new way I've found is to generate chrominance and luminance signals digitally and add them on an analog way after lowpassing. So it's possible to limit the bandwidth of the chrominance and the luminance signals separately.

4.1 chrominance signal

The DDS synthesis and burst generation are like the older versions, but the signal generation is completely redesigned. The component which does this is located in the file **chroma_gen.vhd**.



The necessary phase shift is calculated of the 4 MSB of the carrier signal, rgb and burst. The resolution is limited to 4 bits resulting in phase steps of 22,5 degrees.

- burst has a phase of +135 and -135 degrees, equivalent shift values are 6 and 10 (16-6)
- because of $V=0$ on blue colour the phase shift in this case is 0 deg on odd and even lines.
- the phase shifts of the other colours can be determined by the colour circle, the value is changing between n and $16-n$ between the lines.
- in the case of NTSC generation the phase shift of burst and colour does not alter between the lines

The necessary phase shift is been calculated by a little table and was added to the colour carrier value to get the phase of the modulated carrier. For simplification only the MSB of the addition was used to generate a rectangular signal. For black, white and grey levels the chrominance signal is switched off, for the burst signal a output to generate a reduced amplitude is activated.

This concept is easy expandable to generate more colours in higher phase resolution and/or multiple carrier levels.

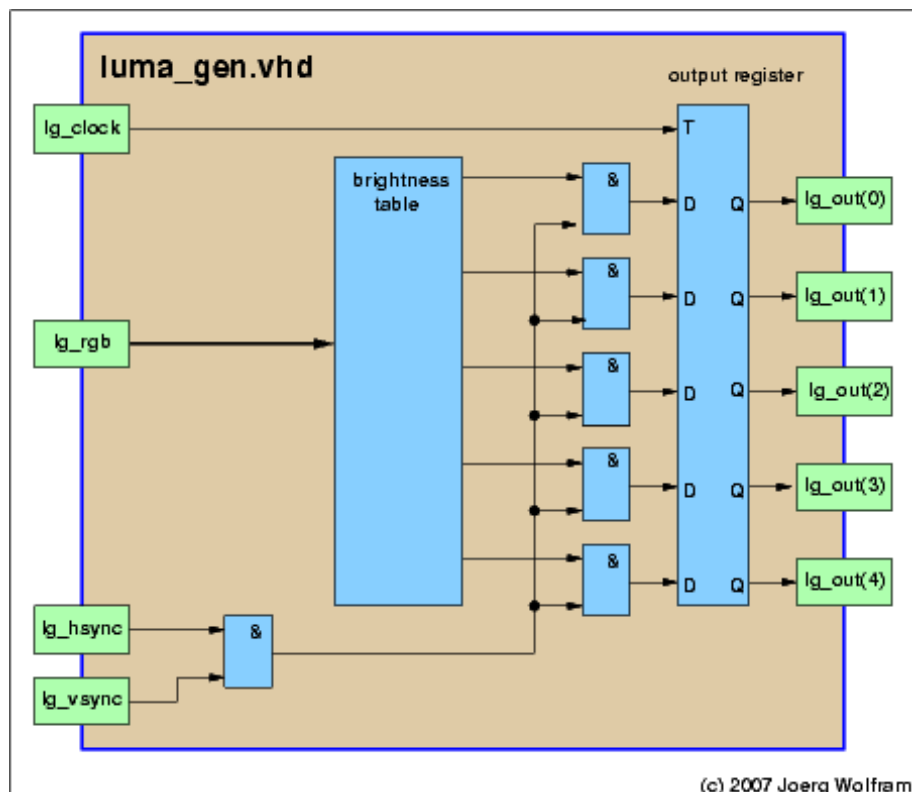
In the result of problems with XST and configurations (ignored by 7.1) in file **main.vhd** the component declaration is followed by the selected architecture. For different clock speeds there are exists 4 architectures:

architecture	system	clock	image quality	CPLD type
clock16	PAL	16 MHz	slightly jitter at colour changes	15ns
clock16	NTSC	16 MHz	highly unuseable (blowing)	15ns
clock20	PAL	20 MHz	slightly jitter at colour changes	10ns
clock20	NTSC	20 MHz	highly unuseable (blowing)	10ns
clock32	PAL	32 MHz	stable image, minimal jitter at colour changes	7ns
clock32	NTSC	32 MHz	stable image, minimal jitter at colour changes	7ns
clock40	PAL	40 MHz	stable image, minimal jitter at colour changes	5ns
clock40	NTSC	40 MHz	stable image, minimal jitter at colour changes	5ns

The testing of the image quality is mostly subjective by using a 5LCD-TV (the only i can use to thest NTSC). At PAL modes the quality is better than at NTSC modes, but this is more problem of the TV standards. The 16- and 20MHz verions are highly unuseable for NTSC generation because of blowing from coloured to non coloured areas in the image. Better lowpassing of the chrominance signal should help but is not tested.

4.2 Luminance signal

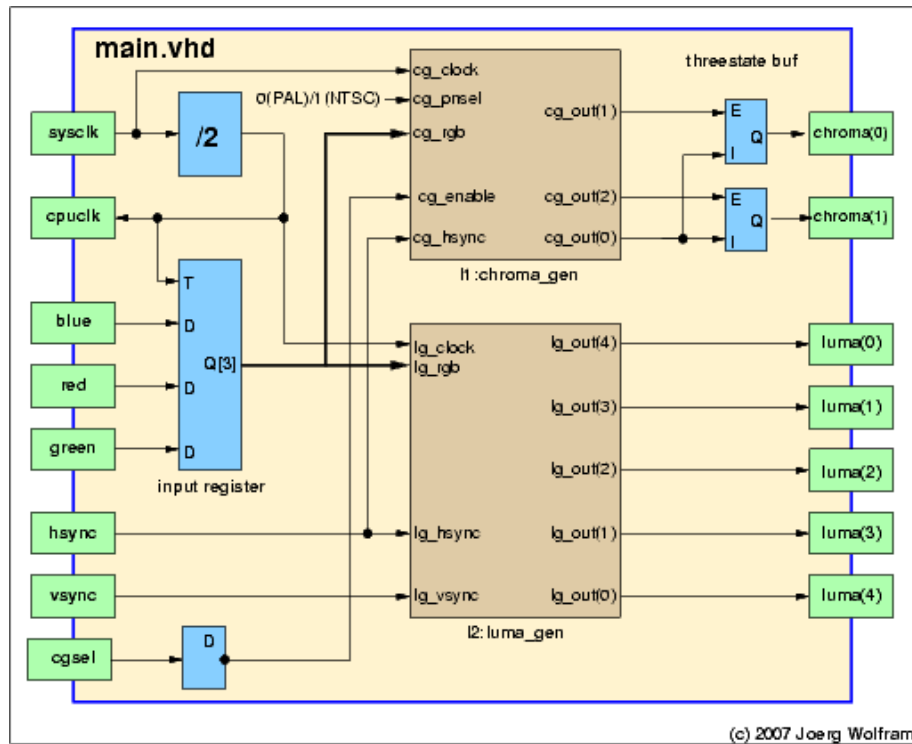
The luminance signal generation is the same as generate B/W only signals. It uses a little table to calculate the different levels. This component is located in file **luma_gen.vhd**.



In some cases it is also possible to generate the luminance signal directly by the rgb source without using this component. For example, if your MCU is already generating an BAS signal you can add the chrominance signal in the simple way of an capacitor.

4.3 The example

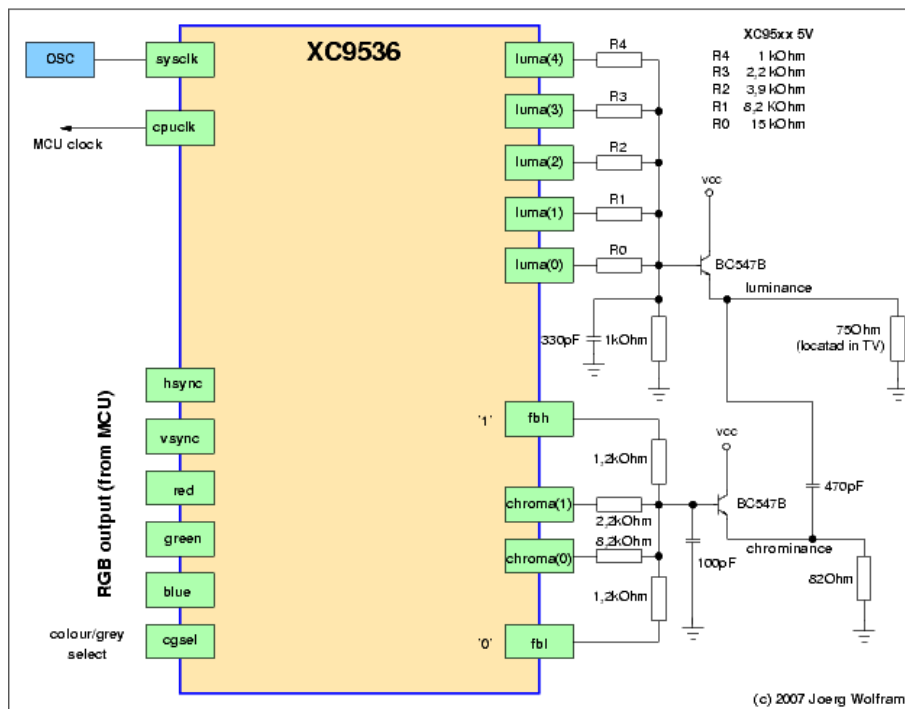
To get a chip, who's working a little bit of additional logic (glue) is necessary:



The rgb signals are stored in registers to get a more synchronous design and to avoid glitches by different input delays. Also 2 threestate outputs for the chrominance-signals (burst and colour) and if needed, a clock divider for the CPU/MCU clock. The luminance signal generation is clocked only by the CPU clock because faster clocking has no advantage and provides undesirable noise. The cgssel signal can be used as a fourth bit to select one of 8 grey levels or with an external switch. '0' selects colour mode and '1' selects greylevel mode. With the cg_pnsel signal it is possible to switch between PAL ('0') and NTSC ('1') without any hardware changes. If only PAL or NTSC is needed, feeding this signal with statically '0' or '1' results in decreasing the amount of logic cells.

4.4 Connecting to the world outside

This part was completely redesigned due to variant output impedances of different CPLD/FPGA and to reduce the moving pattern from older versions.



The two transistors are used for impedance transformation so the output impedance has not more much effect to the result. In the upper part the luminance signal is generated and in the lower part the chrominance signal. The two 1,2KOhm resistors on **fbh** and **fbl** pin sets the chrominance output to a defined level. The chrominance signal is mixed with the luminance signal by an simple capacitor, this is also used in cheap s-video to FBAS adapters. So this works, it should be easy to create s-video by splitting this connection, but this feature is untested.

5 Timing

Due to the complex logic, the timing in chroma-path is a little bit critical and limits clock speed. For correct results for 32MHz a 7ns CPLD is needed, for 40MHz a 5ns one. So fast spikes are filtered by lowpassing at the output, a 15ns Device works well in all cases. For better timing more macrocells (38-40) are necessary to flatten the design. Another way is to store stages of the chrominance signal generation in stages (pipelining). This causes in delay between chrominance and luminance signals which should not be visible on TV. Or, the luminance signal must also beeing delayed. For PAL and a little bit less of image quality the 16- and 20MHz versions are an good alternative.

6 Result

The design written in VHDL fits in a cheap XC9536 CPLD if only PAL or NTSC signals are generated. Walking pattern visible on some TV at older versions of the design are minimized. The design is more easy expandable but by the cost of an increased amount of external hardware. For some cases ther are ready-to-use jedec files in the **jedec** folder.

7 Changelog

17.2.2007 the first public version (0.21)

- PAL and version NTSC for 16, 20, 32 and 40 MHz clock frequency

9.3.2007 version (0.31)

- give up the way with a long ROM table in favor of an mixed signal generation
- a more clearly design and improved image quality