PT56

Advanced Composite Video Interface:

Encoder IP Core

User Manual

Revision 0.1
29th February 2020

Revisions

Date	Revisions	Version
29-02-2020	First Draft.	0.1

Contents

Tables

Figures

1. Introduction

PT56 is an aCVi®[1] encoder IP (intellectual property) core compatible with the aCVi® Advanced Composite Video Interface (revision 2).

aCVi is a method to transmit high definition video over long distances (>300m) of low cost coaxial or twisted-pair cable. SingMai are proposing aCVi as an open standard. The current version of the standard may be found here:
https://www.singmai.com/Documents/aCVi%20Format%20Specification%200.1.pdf

The encoder IP accepts either separate YCbCr 4:2:2 digital component data or RAW data direct from an image sensor, together with the associated video clock (74.25MHz) and horizontal and vertical timing signals, which it encodes to a single 10 bit straight 2's complement composite output for driving a suitable DAC (digital to analogue converter) and amplifier. PT56 currently supports 720p-25Hz/30Hz/50Hz/59.94Hz/60Hz and 1080p-24Hz/25Hz/29.97Hz/30Hz HD video formats.

Control and status registers are written to and read from using a conventional 8-bit wide microprocessor interface.

The intellectual property block is provided as RTL compliant Verilog-2001 source code for FPGAs from all vendors or for ASICs.

Typical resource usage for an Altera FPGA is shown in Table 1.

Logic Cells	Memory Bits	M9K blocks	9x9 Multipliers	18x18 multipliers

Table 1 PT56 Altera FPGA resource requirements

An approximate equivalent for ASIC resource usage is TBA LCs (logic cell only compile for Altera FPGA) x 14 ~ TBA 2 input NAND gate equivalent. The memory is 13kb of single port ROM (512 x 24).

[1] aCVi® is a registered trademark of SingMai Electronics Ltd.

2. PT56 Module description

The PT56 aCVi encoder IP core comprises 7 Verilog modules in a hierarchical structure, (see Table 2).

PT56_encoder.v	Register_control.v	
	Cin.v	
	Yin.v	
	SPG.v	
	Modulator.v	SinCos_ROM.v

Table 2 PT56 Verilog file structure.

The top level file is PT56_encoder.v which, in turn, calls seven of the other modules. Modulator.v calls a third level file, SinCos_ROM.v.

3. Signal Interconnections

The PT56 signal interconnect diagram is shown in Figure 1.

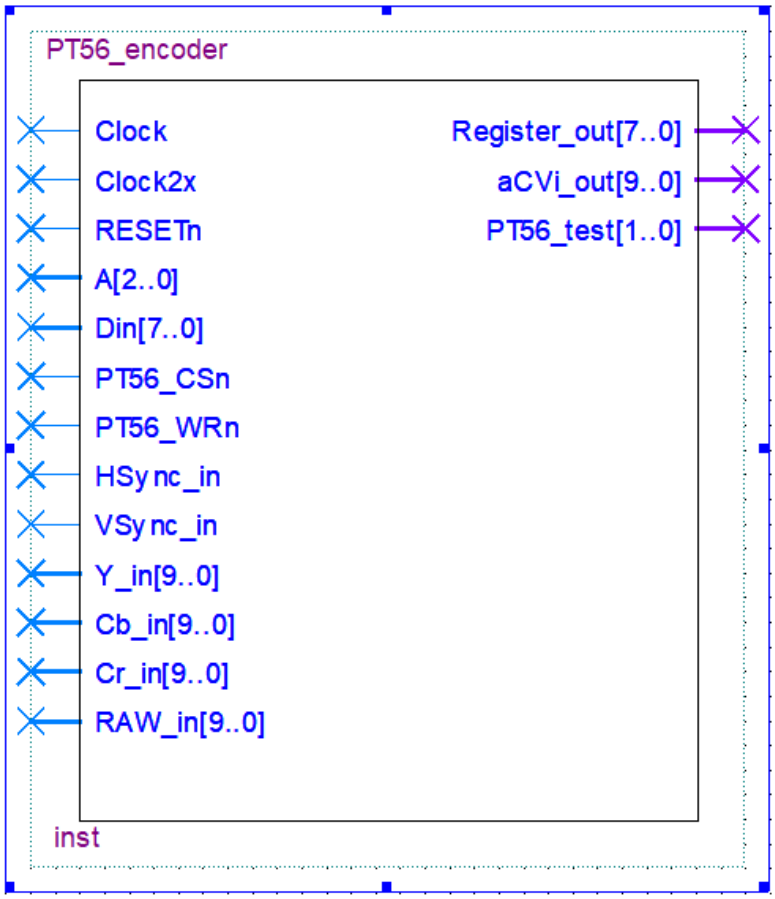

Figure 1 PT56 Block schematic.

The signal descriptions are shown in Table 3, below.

Inputs	
Signal	**Description**
Clock	Pixel clock input (74.25MHz/ 74.17582418MHz). All video data inputs should be valid at the rising edge of this clock.
Clock2x	Twice the Clock input frequency (148.5MHz/148.3516484MHz). Output data is valid on the rising edge of this clock. Rising edges of Clock and Clock2x should be coincident.
RESETn	Asynchronous active low reset signal. Asserting this input sets all the control registers to their default value and resets all registers.
A[2:0]	Control address bus input used to select the control register to be written to/read from.
Din[7:0]	Control data input bus.
PT56_CSn	Control chip select input, active low. Used in combination with the WRn input to control writing to the control registers.

PT56_WRn	Active low write enable input. Used in combination with the CSn input to control writing to the control registers.
HSync_in	Horizontal synchronization input (e.g. for 720p/60Hz operation this input is at 45kHz). Active low input, the falling edge is the 0H timing reference point. This input must be at least 4 'Clock' periods wide. (See input formatting on p.12).
VSync_in	Vertical synchronization input (e.g. for 720p/60Hz operation this input is at 60Hz). Active low input, the falling edge is the 0V timing reference point. This input must be at least 4 'Clock' periods wide. (See input formatting on p.12).
Y_in[9:0]	Y (luma) input to the encoder. The input is straight binary, blanking level is 64_{10} and peak level 960_{10}. The data input should be valid at the rising edge of 'Clock'. Y_in[9] is the MSB. If the input is 8-bits wide, the bottom 2 bits should be tied to '0'. (See input formatting on p.13).
Cb_in[9:0]	Cb (B-Y chroma) input input to the encoder. The input is offset binary, blanking level is 512_{10}. The data input should be valid at the rising edge of 'Clock'. Cb_in[9] is the MSB. If the input is 8-bits wide, the bottom 2 bits should be tied to '0'. (See input formatting on p.13).
Cr_in[9:0]	Cr (R-Y chroma) input to the encoder. The input is offset binary, blanking level is 512_{10}. The data input should be valid at the rising edge of 'Clock'. Cr_in[9] is the MSB. If the input is 8-bits wide, the bottom 2 bits should be tied to '0'. (See input formatting on p.13).
RAW_in[9:0]	Multiplexed RGB data from an image sensor. The data input should be valid at the rising edge of 'Clock'. RAW_in[9] is the MSB. See input formatting chapter for information on the demultiplexing of the RAW input.
Outputs	
Signal	**Description**
Register_out[7:0]	Control output data bus. Outputs the control/status register data selected by the A[2:0] bus.
aCVi_out[9:0]	Encoded aCVi video output data. aCVi_out[9] is the MSB. The output is 2's complement coded and is valid at the rising edge of 'Clock2x'.

Table 3 PT56 Input/Output signals

The Verilog instantiation of PT56 is shown below:

```
PT56_encoder PT56_encoder_inst
(
.Clock(Clock_sig) ,                    // input  Clock_sig
.Clock2x(Clock2x_sig) ,                // input  Clock2x_sig
.RESETn(RESETn_sig) ,                  // input  RESETn_sig
.A(A_sig) ,                            // input [2:0] A_sig
.Din(Din_sig) ,                        // input [7:0] Din_sig
.PT56_CSn(PT56_CSn_sig) ,              // input  PT56_CSn_sig
.PT56_WRn(PT56_WRn_sig) ,              // input  PT56_WRn_sig
.HSync_in(HSync_in_sig) ,              // input  HSync_in_sig
.VSync_in(VSync_in_sig) ,              // input  VSync_in_sig
.Y_in(Y_in_sig) ,                      // input [9:0] Y_in_sig
.Cb_in(Cb_in_sig) ,                    // input [9:0] Cb_in_sig
.Cr_in(Cr_in_sig) ,                    // input [9:0] Cr_in_sig
.RAW_in(RAW_in_sig) ,                  // input [9:0] RAW_in_sig

.Register_out(Register_out_sig) ,      // output [7:0] Register_out_sig
.aCVi_out(aCVi_out_sig) ,              // output [9:0] aCVi_out_sig
.PT56_test(PT56_test_sig)              // output [1:0] PT56_test_sig
);
```

4. aCVi Overview

The following is a brief overview of the aCVi® revision 2 interface (abbreviated to aCVi® in this document).

aCVi® is a proprietary format, developed by SingMai Electronics, to transmit high definition video over long distances of coaxial or twisted pair cable. aCVi® is an update to the previous version, specifically designed to interface directly to image sensors, although it may also be used to transmit conventional video sources.

A single chip image sensor, as found in almost all non-broadcast cameras, uses a colour filter to 'assign' each sensor pixel one of red, green or blue sensitivities. Because green is where the human eye is most sensitive, there are twice as many green pixels as red and blue (see Figure 2). This means that if your sensor has a horizontal array of 1920 pixels, only 960 of them are green, red or blue pixels, and for the red and blue pixels, each horizontal line is either red or blue. The actual resolution of the sensor to each colour is for green, 960 x 1080 pixels, and for red and blue, 960 x 540 pixels. (A broadcast camera will use three optically aligned sensors, each offering 1920 x 1080 pixels for the three colours). If we refer to the full resolution (e.g. a broadcast camera) as 4:4:4 sampled, a single image sensor actually produces a 2:2:0 output.

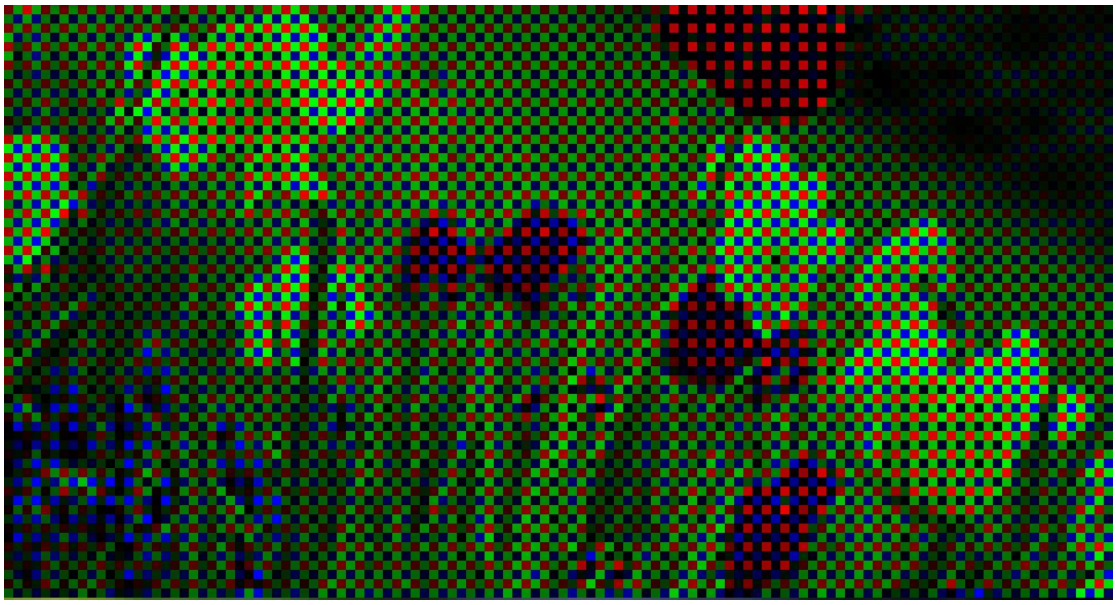

Figure 2 Bayer colour filter.

To conform with video standards (e.g. 1920 x 1080) the additional pixels are interpolated (a technique known as Bayer de-mosaicing) and this function is usually performed in the camera ISP (Image Signal Processor). However this process can produce artifacts into the image (for example see the colour artifacts on the white fence in Figure 3), and also, because it generates more than double the amount of original pixels, more than doubles the bandwidth of the output signal, which exacerbates the problem if the video is required to transmitted long distances.

aCVi® interfaces directly to the single chip image sensor and transmits the RAW 2:2:0 resolution image directly, thereby reducing by more than half the bandwidth of the transmitted signal and achieving higher resolution, lower noise and greater distances.

Figure 3 Left: Original full resolution image. Right: Image after Bayer demosaicing.

The available aCVi® IP cores are shown in Figure 4. The PT56 provides an aCVi® transmitter for RAW or YCbCr (BT1120 style interface) and outputs digital aCVi for a digital to analogue converter (DAC).

The receiver IP core is the PT52. An analogue front end (AFE) conditions the analogue aCVi video before it is converted to digital aCVi in an analogue to digital converter (ADC). The PT52 decodes the aCVi input (RAW or YCbCr format) into a YCbCr output.

Figure 4 aCVi IP cores.

5. Technical Overview

A simplified block diagram of the PT56 encoder is shown in Figure 3.

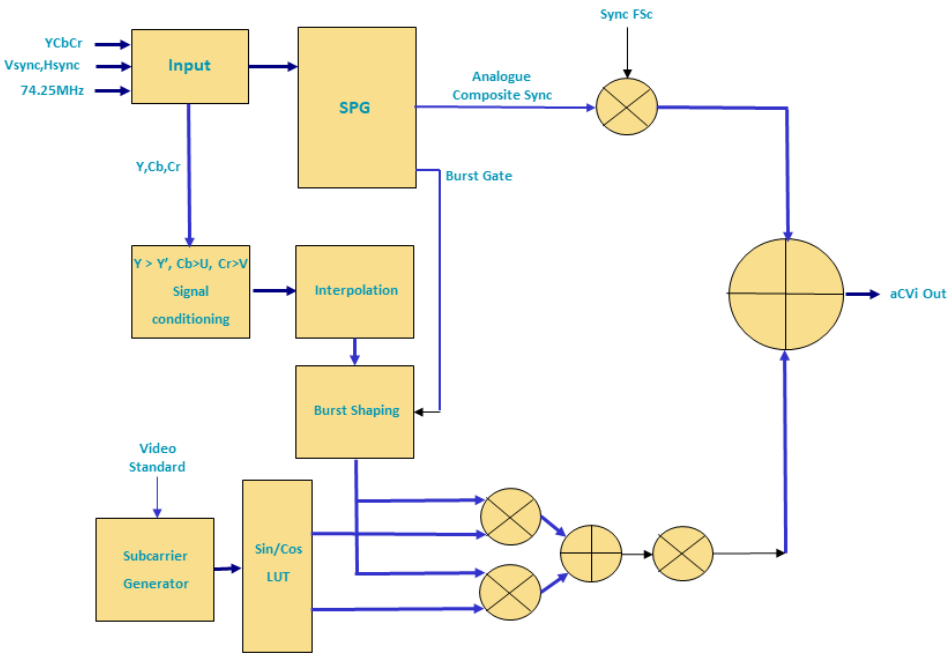

Figure 5 PT56 Block diagram.

PT56_encoder.v

This is the top-level design file and it interconnects all the following modules.

Register_control.v

A conventional 8-bit microprocessor style control interface is used to write and read to the PT56 control registers. Details of the interface may be found in Chapter 8 and the register descriptions may be found in Chapter 9.

Yin.v

The Y_in module accepts the luma data from the YCbCr input or the demultiplexed green channel from the RAW input. The video data is interpolated from 74.25MHz to 148.5MHz using a 63-tap FIR filter – the frequency response of this filter is shown in Figure 6. The interpolated luma/green video is then sent to the Modulator.v module.

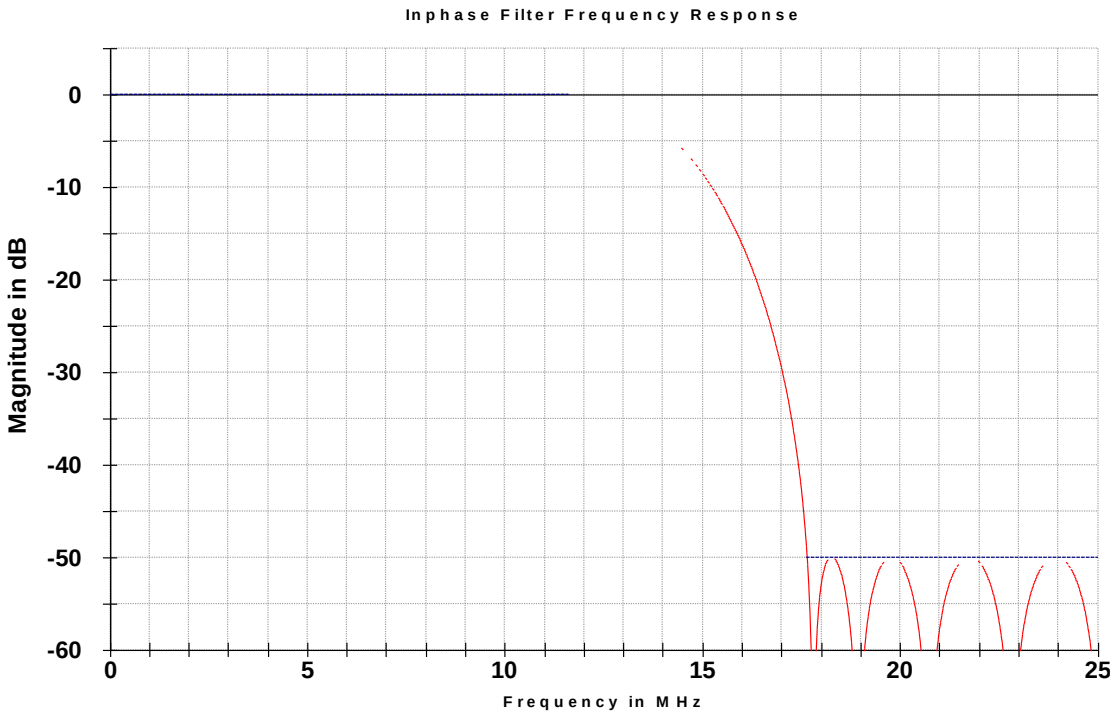

Figure 6 Interpolation filter.

Cin.v

The C_in module accepts either the Cb/Cr data from the YCbCr input, or the red/blue input from the RAW input. The RAW data input is already line multiplexed (i.e. one line is red only video, the second line is blue only video). For the YCbCr input, the Cb/Cr inputs are line multiplexed using the half-line signal from the SPG. For both RAW and YCbCr inputs the multiplexing sequence is synchronized to the vertical sync timing.

The line-multiplexed video is then interpolated to 148.5MHz using the same filter design as used for the luma/green channel (see Figure 6). The interpolated Cb-Cr/red-blue video is then sent to the Modulator.v module

SPG.v

HSync_in (horizontal) and VSync_in (vertical field) signals are used for picture synchronization. The currently supported video standards are shown in Table 4.

The falling edge of the horizontal pulse input is used to reset a 12-bit counter clocked at 'Clock' frequency. This counter is used to generate a delay which may be programmed (HPhase). This programmable delay creates the 0H horizontal timing reference from which all the other timing pulses are generated (see Figure

Standard	Pixels/line	Line frequency	Clock frequency	Horizontal sync width (clocks)	Broad pulse start position (clocks)	Broad pulse end position (clocks)	Sync Subcarrier frequency
720p/25Hz	3960	18.750kHz	74.25MHz	80	296	3590	10.561875MHz
720p/30Hz	3300	22.500kHz	74.25MHz	80	296	2930	10.51425MHz
720p/50Hz	1980	37.500kHz	74.25MHz	80	296	1610	10.39875MHz

720p/59Hz	1650	44.955kHz	74.18MHz	80	296	1280	10.488012MHz
720p/60Hz	1650	45.000kHz	74.25MHz	80	296	1280	10.4985MHz
1080p/24Hz	2750	27.000kHz	74.25MHz	80	236	1920	10.5111MHz
1080p/25Hz	2640	28.125kHz	74.25MHz	80	236	1920	10.4990625MHz
1080p/29Hz	2200	33.716kHz	74.18MHz	80	236	1920	10.49587912MHz
1080p/30Hz	2200	33.750kHz	74.25MHz	80	236	1920	10.506375MHz

Table 4 aCVi sync formats.

The Hphase setting is therefore used to align the active video input with the output horizontal timing signals.

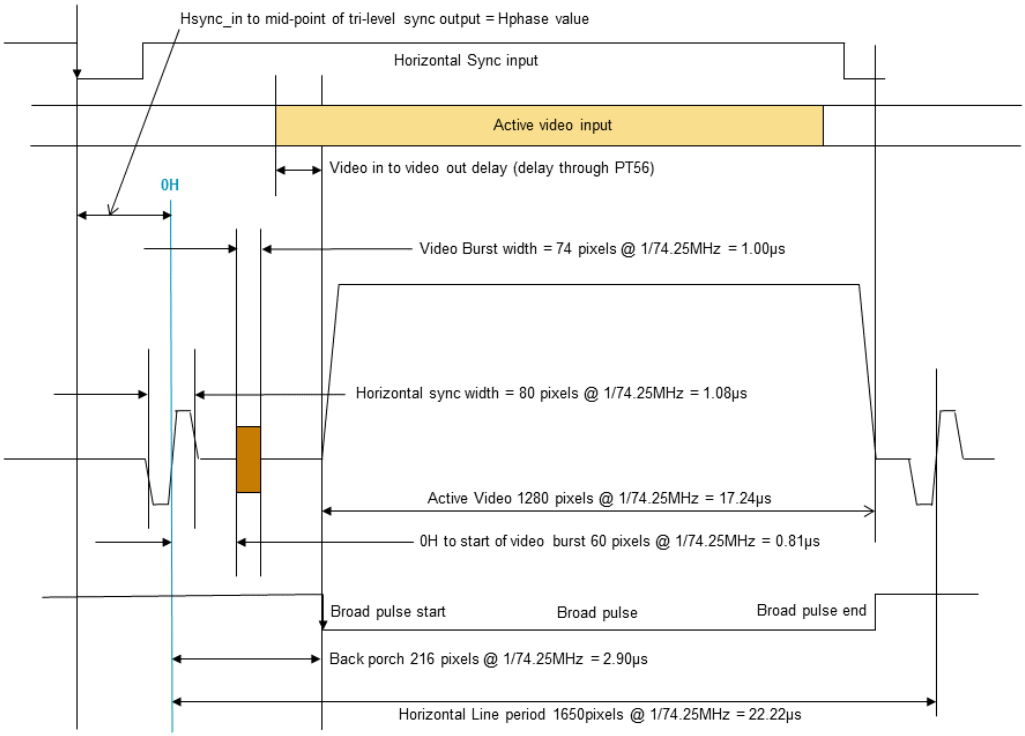

Figure 7 PT56 horizontal timing (720p/60Hz timing).

Similarly, the falling edge of the vertical pulse input is used to reset an 11-bit counter clocked at the beginning of each horizontal line (e.g. a line counter). This counter (VPhase) generates the delay for the vertical sync generation, which consists of five broad pulses (see Figure 8). The vertical and horizontal sync pulses are combined and then shaped in a raised cosine filter. This composite sync pulse is then sent to the Modulator.v module.

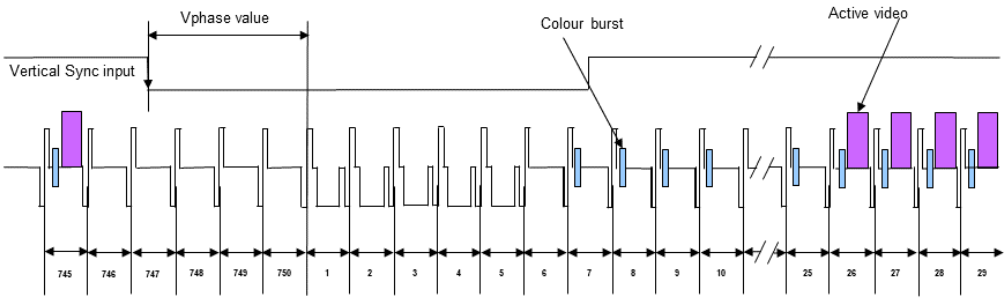

Figure 8 PT56 vertical timing (720p).

Modulator.v

A 32-bit ratio counter clocked from the 148.5MHz clock generates the sync pulse subcarrier frequency, as detailed in Table 4.

The top 11 bits of this ratio counter (the phase word) address a ROM containing sine and cosine values. These waveforms are multiplied by the analogue composite sync input from the SPG.v module and then added together. The sine channel is used as a reference and the cosine channel becomes the sync channel.

A similar technique is used to modulate the video. The ratio counter generates the video subcarrier frequency shown in Table 5. The luma from the Y_in.v module modulates the sine channel and the line-multiplexed chroma modulates the cosine channel. The two channels are added together. A 74-clock wide sample (burst gate) of the sine channel is added as a reference (colour burst) for the video demodulator.

Standard	Pixels/line	Line frequency	Clock frequency	Burst gate start position (clock periods)			Video Subcarrier frequency
720p/25Hz	3960	18.750kHz	74.25MHz	151			19.509375MHz
720p/30Hz	3300	22.500kHz	74.25MHz	151			19.49625MHz
720p/50Hz	1980	37.500kHz	74.25MHz	151			19.51875MHz
720p/59Hz	1650	44.955kHz	74.18MHz	151			19.488012MHz
720p/60Hz	1650	45.000kHz	74.25MHz	151			19.5075MHz
1080p/24Hz	2750	27.000kHz	74.25MHz	121			19.5075MHz
1080p/25Hz	2640	28.125kHz	74.25MHz	121			19.50487MHz
1080p/29Hz	2200	33.716kHz	74.18MHz	121			19.50487MHz
1080p/30Hz	2200	33.750kHz	74.25MHz	121			19.490625MHz

Table 5 Video modulator parameters.

The modulated sync, the modulated luma/chroma and the colour burst are then added together to form the final aCVi waveform (see Figure 9).

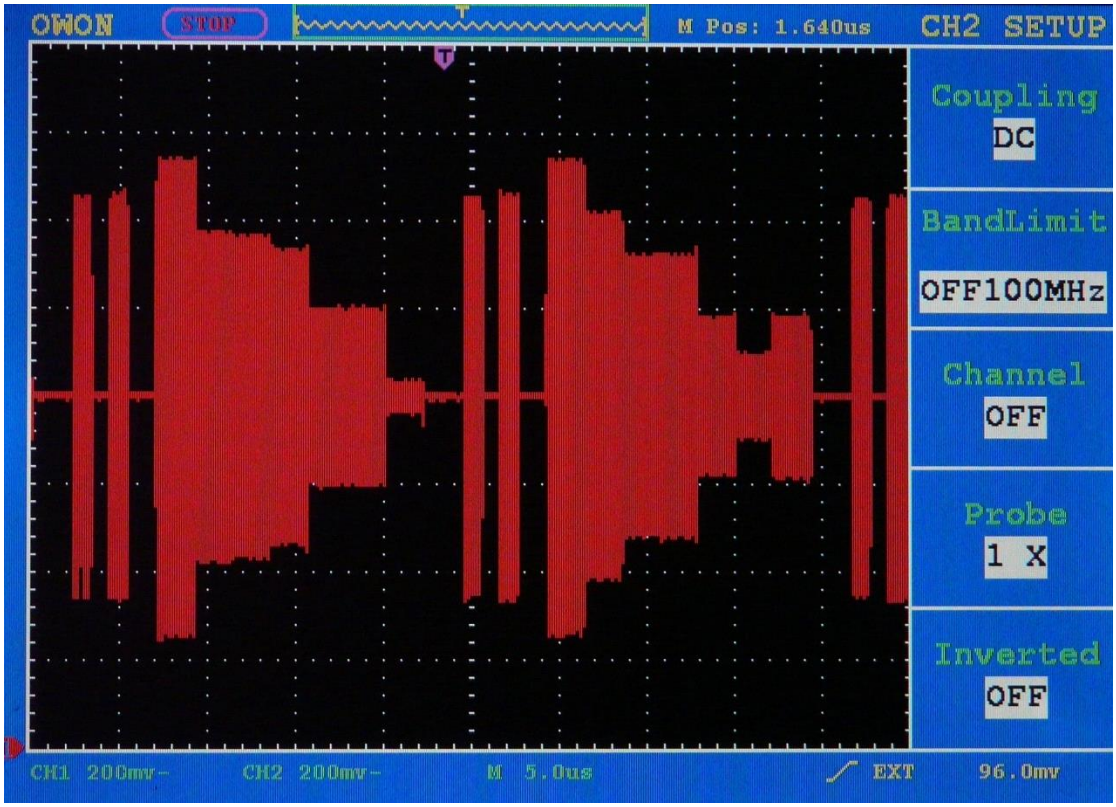

Figure 9 aCVi waveform (75% colour bars).

6. Data Transfers

TBD.

7. Register interface

Figure 10 shows the timing diagram for the register interface; it is a conventional microprocessor interface. Each register is selected via a 5-bit address bus. Writes to unused register locations are ignored.

To write to the selected register the PT56_CSn (chip select) input must be asserted low and the A[4:0] register address and the data for this register set up. The PT56_WRn input must then be driven low and high again: On the rising edge of this pulse the data is latched into the address selected. The PT56_CSn input should then be returned high.

For the write to occur reliably the address (A[4:0]) and data (Din[7:0]) must be stable and valid during the low to high transition of the PT56_WRn pulse.

The address input also selects the register data that is presented on the Register_out[7:0] bus. This output is independent of the PT56_CSn or PT56_WRn inputs.

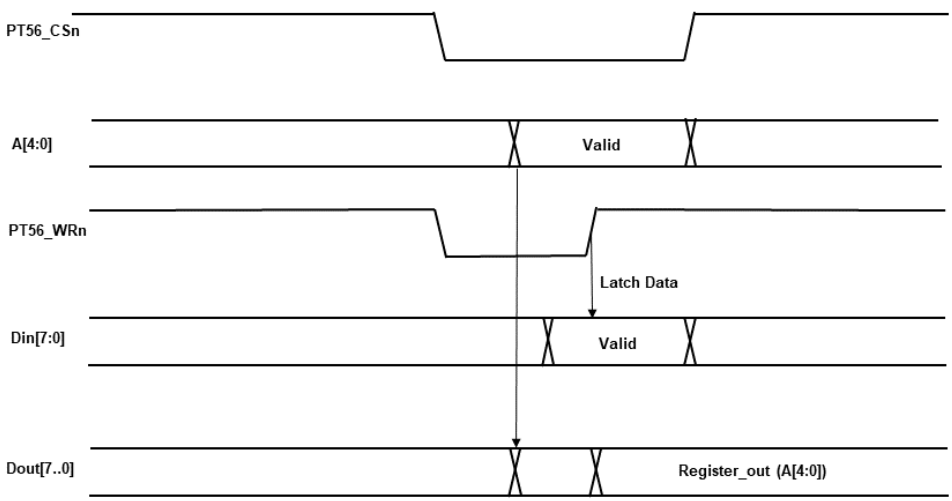

Figure 10 PT56 Register control.

8. Register descriptions

Table 8 lists all of the control and status registers. All of the registers are 8-bit; unused register bits read back as zeros.

Please note that some registers can be set to values that are illegal and will produce invalid outputs.

Asserting the RESETn input sets the PT56 registers to their default values.

Register Offset	Register Name	R/W	Bit Value	Description	
Control Registers					
$00	**Control 1**	R/W		PT56 control 1 (video standard)	
			7	Not used.	
			6:0	Video standard	
				Video standard[6:0]	Standard
				0000	720p25
				0001	720p30
				0010	720p50
				0011	720p59
				0100	720p60
				0101	1080p24
				0110	1080p25
				0111	1080p29
				1000	1080p30
$01	**Control 2**	R/W		PT56 control 2	
SPG					
Output					

Table 6 Register Descriptions.

9. Output Interface

The output of the aCVi encoder is 10-bit, 2's complement, video data at 148.5MHz.

This has to be converted to analogue using a digital to analogue converter. On the evaluation board (SM06 revision 0.5) an Analog Devices 10-bit DAC, the AD9705, is used for this purpose.

The FPGA also provides a 148.5MHz clock for the DAC.

The differential analogue outputs from the DAC (1.0V pk-pk) are then amplified by U21 and filtered (U9) to remove clock noise and driven through a cable impedance matching series resistor to the coaxial or twisted-pair cable.

The schematics for this are shown in Figure 11.

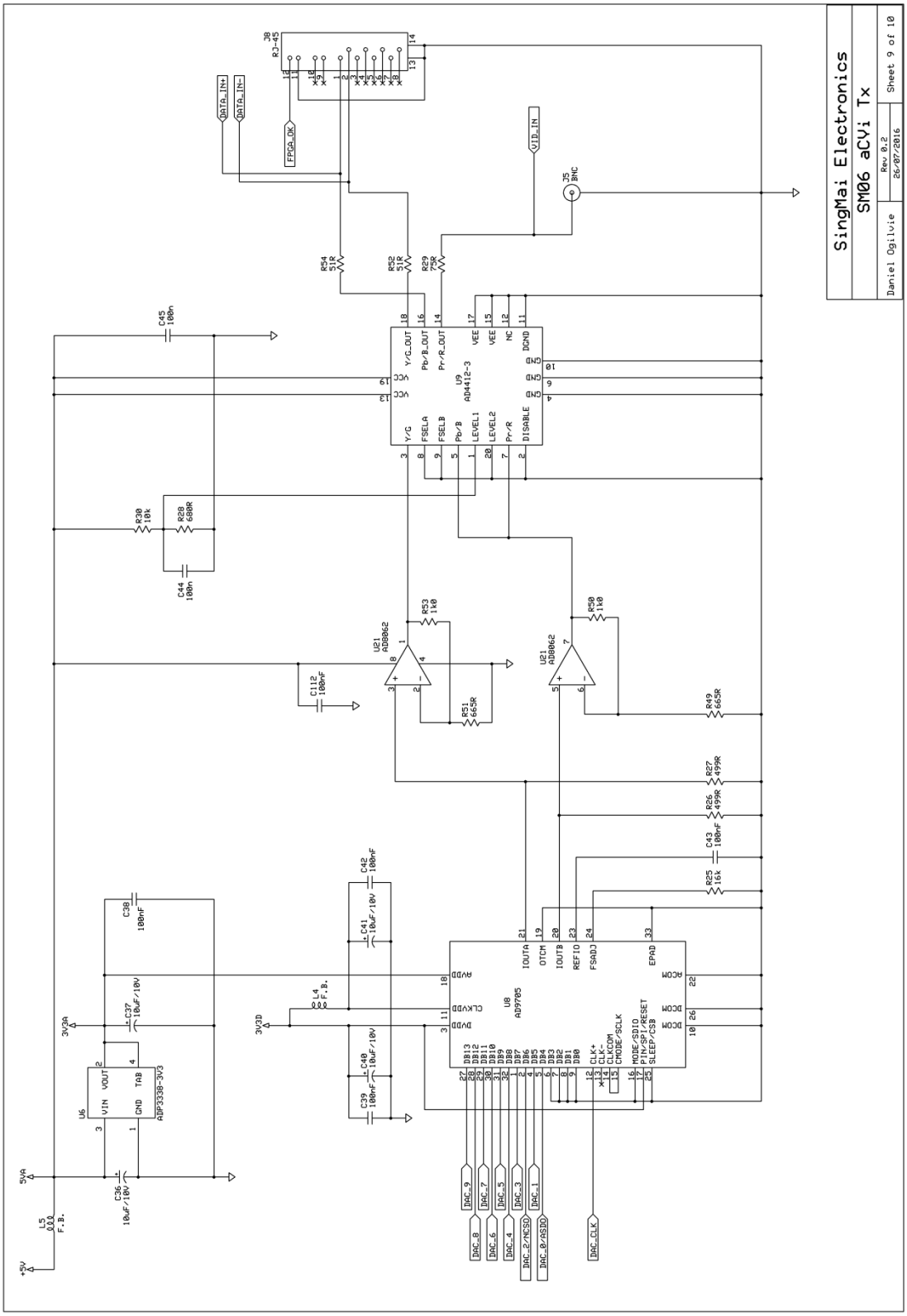

Figure 11 PT56 output interface schematic.