PT57

HD video
Character/Text Overlay

User Manual

Revision 0.1
19th April 2017

Revision History

Date	Revisions	Version
19-04-2017	First Draft.	0.1

Contents

Tables

Figures

1. Introduction

PT57 is a character/text based overlay for high definition video.

PT57 offers 112 pre-programmed characters which can be inserted into an HD video source. The PT57 character generator uses just 24k of memory (single port ROM) and the character memory can be programmed for just the size required being programmable for both number of horizontal characters and number of rows. Each character is 12x16 pixels. The colour of the character and the background is programmable.

PT57 supports 720p/25,30,50,59.94,60Hz; 1080p/24,25,29.97,30Hz and 1080i/50,59.94,60Hz standards.

PT57 is ideal for inserting text, channel IDs, menus, status and titles while using minimal FPGA/ASIC resources.

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 (as compiled for an EP4CE15 FPGA used on the SM08 PT52 evaluation board).

Logic Elements	Memory Bits	M9K blocks	9x9 Multipliers	18x18 multipliers
272	24576 + RAM	3	0	0

Table 1 PT57 Altera FPGA resource requirements

An approximate equivalent for ASIC resource usage for the logic core is 4000 2 input NAND gate equivalent. The memory requirement is 24576 bits of single port ROM for the character ROM and the character RAM is calculated by the number of characters (e.g. 64 characters/line x 8 rows = 512 words) x 7 bits depth.

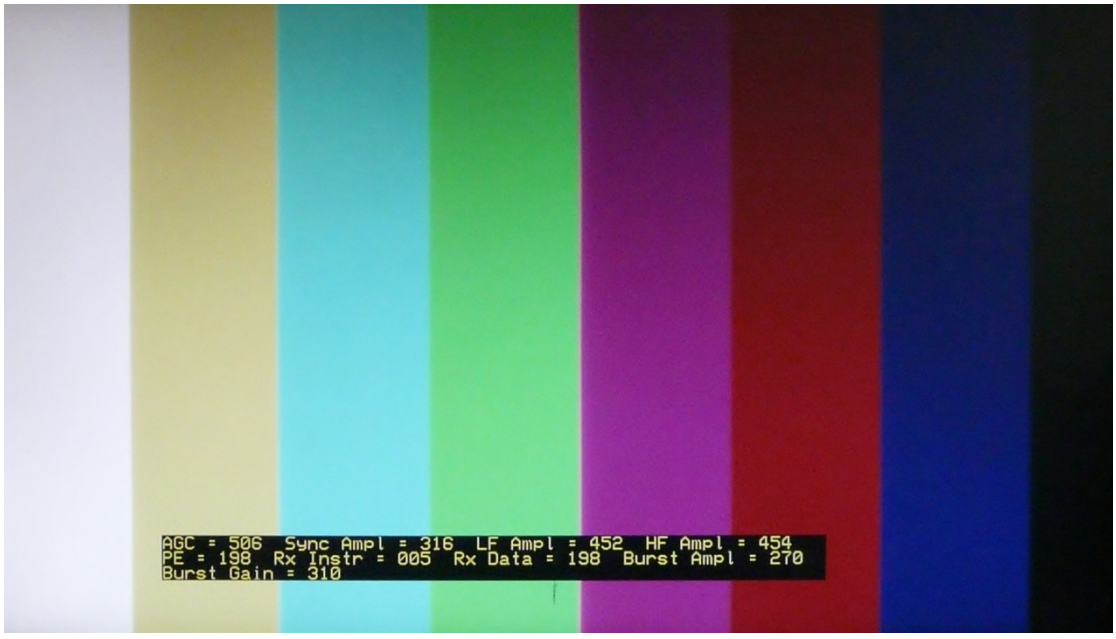

Figure 1 Example of the PT57 used to provide status information.

2. Signal Interconnections

The PT57 signal interconnect diagram is shown in Figure 2.

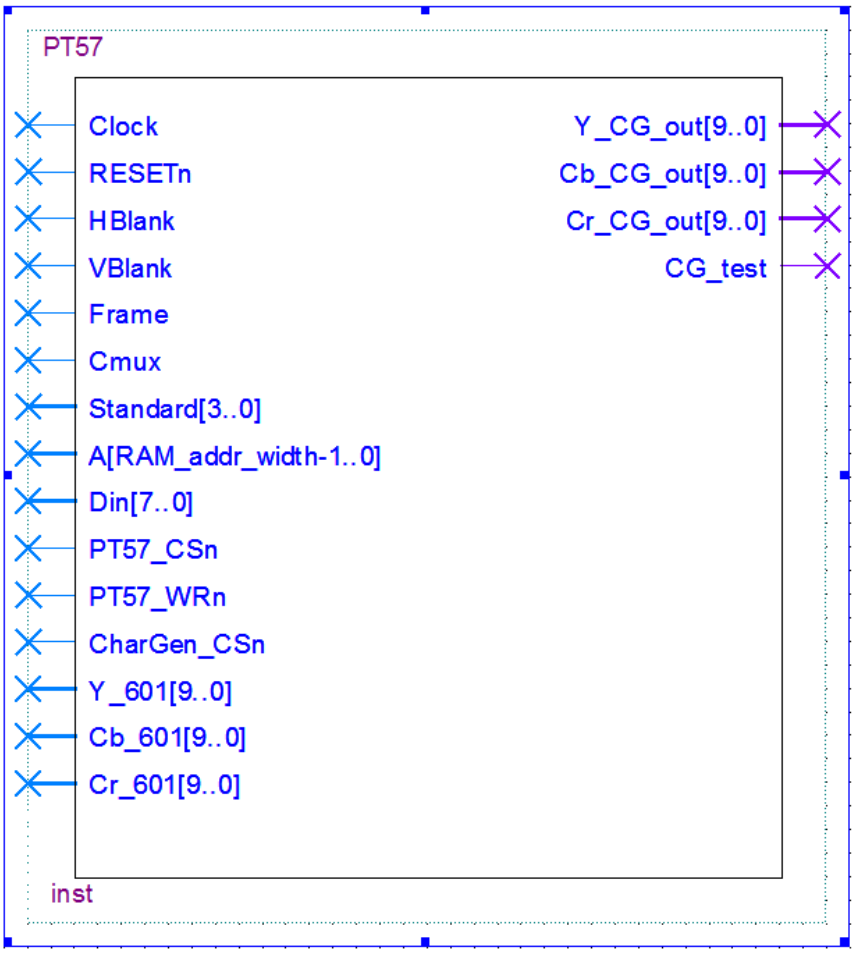

Figure 2 PT57 Interconnections.

The signal descriptions are shown in Table 2, below.

Signal	Description
Clock	74.25MHz clock input. Video and synchronizing inputs should be stable at the rising edge of this clock.
RESETn	Asynchronous active low reset signal. Asserting this input sets all the control registers to their default value and resets all registers.
HBlank	Horizontal blanking input (high during blanking, low during active video). The high to low transition of this input is used to start the left margin counter. This input should be valid on the rising edge of 'Clock'.
VBlank	Vertical blanking input (high during blanking, low during active video). The high to low transition of this input is used to start the top margin counter. This input should be valid on the rising edge of 'Clock'.
Frame	If using an interlaced video standard this input indicates odd or even field. If using progressive inputs this input should be set to '0'. This input should be valid on the rising edge of 'Clock'. (Note: interlaced inputs need Control register 1, bit 0 to be set to '1'.)
Cmux	If the Y, Cb, Cr video inputs are 4:2:2 format, this input is the enable for the Cb and Cr inputs and should be at 74.25/2 = 37.125MHz. If the video input is 4:4:4 this input should be set to '1'. This input should be valid on

	the rising edge of 'Clock'.
Standard[3:0]	These DC inputs indicate the video standard (see Table 3).
A[RAM_addr_width:0]	Control address bus input used to select the register or character memory location to be written to/read from.
Din[7:0]	Control/character memory data input bus.
PT57_CSn	Control chip select input, active low. Used in combination with the PT57_WRn input to control writing to the registers.
PT57_WRn	Active low write enable input. Used in combination with the PT57_CSn and CharGen_CSn inputs to control writing to the registers and character memory.
CharGen_CSn	Character memory chip select input, active low. Used in combination with the PT57_WRn input to control writing to the character memory.
Y_601[9:0]	Luma input to the PT57. It is expected this input is straight binary, with a blanking level of 64_{10} and a white amplitude of 940_{10}. This input should be valid on the rising edge of 'Clock'.
Cb_601[9:0]	Cb (Chroma) input to the PT57. It is expected this input is offset binary, with a blanking level of 512_{10} This input should be valid on the rising edge of 'Clock'.
Cr_601[9:0]	Cr (Chroma) input to the PT57. It is expected this input is offset binary, with a blanking level of 512_{10} This input should be valid on the rising edge of 'Clock'.
Y_CG_out[9:0]	Luma output from the PT57. This output is a copy of the Y_601[9:0] input (with a 1 clock delay) with the character overlaid (if enabled). This output is valid on the rising edge of 'Clock'.
Cb_CG_out[9:0]	Cb (chroma) output from the PT57. This output is a copy of the Cb_601[9:0] input (with a 1 clock delay) with the character overlaid (if enabled). This output is valid on the rising edge of 'Clock' when Cmux is '1'.
PT2_test[1:0]	Cr (chroma) output from the PT57. This output is a copy of the Cr_601[9:0] input (with a 1 clock delay) with the character overlaid (if enabled). This output is valid on the rising edge of 'Clock' when Cmux is '1'.
CG_test	Test output. Do not connect.

Table 2 Input/Output signals

The Verilog instantiation of PT57 is shown below:

```
PT57 PT57_inst
(
.Clock(Clock_sig),                          // input  Clock_sig
.RESETn(RESETn_sig),                        // input  RESETn_sig
.HBlank(HBlank_sig),                        // input  HBlank_sig
.VBlank(VBlank_sig),                        // input  VBlank_sig
.Frame(Frame_sig),                          // input  Frame_sig
.Cmux(Cmux_sig),                            // input  Cmux_sig
.Standard(Standard_sig),                    // input [3:0] Standard_sig
.A(A_sig),                                  // input [RAM_addr_width-1:0] A_sig
.Din(Din_sig),                              // input [7:0] Din_sig
.PT57_CSn(PT57_CSn_sig),                    // input  PT57_CSn_sig
.PT57_WRn(PT57_WRn_sig),                    // input  PT57_WRn_sig
.CharGen_CSn(CharGen_CSn_sig),              // input  CharGen_CSn_sig
.Y_601(Y_601_sig),                          // input [9:0] Y_601_sig
.Cb_601(Cb_601_sig),                        // input [9:0] Cb_601_sig
.Cr_601(Cr_601_sig),                        // input [9:0] Cr_601_sig

.Y_CG_out(Y_CG_out_sig),                    // output [9:0] Y_CG_out_sig
```

```
.Cb_CG_out(Cb_CG_out_sig) ,          // output [9:0] Cb_CG_out_sig
.Cr_CG_out(Cr_CG_out_sig) ,          // output [9:0] Cr_CG_out_sig
.CG_test(CG_test_sig)                // output  CG_test_sig
);

defparam PT57_inst.no_Hchar = 64;
defparam PT57_inst.H_addr_width = 6;
defparam PT57_inst.no_rows = 4;
defparam PT57_inst.V_addr_height = 2;
defparam PT57_inst.RAM_addr_width = 8;
```

Defparam descriptions.

.no_Hchar	The number of horizontal characters per line.
.H_addr_width	ceil($\log_2$[no_Hchar]). e.g. if the number of horizontal characters is 64, H_addr_width = 6.
.no_rows	The number of horizontal rows of characters.
.V_addr_height	ceil($\log_2$[no_rows]). e.g. if the number of rows is 16, V_addr_height = 4.
.RAM_addr_width	H_addr_width + V_addr_height. e.g. using the above examples, RAM_addr_width = 6 + 4 = 10.

3. Technical Overview

A simplified block diagram of the PT57 character generator is shown in Figure 3.

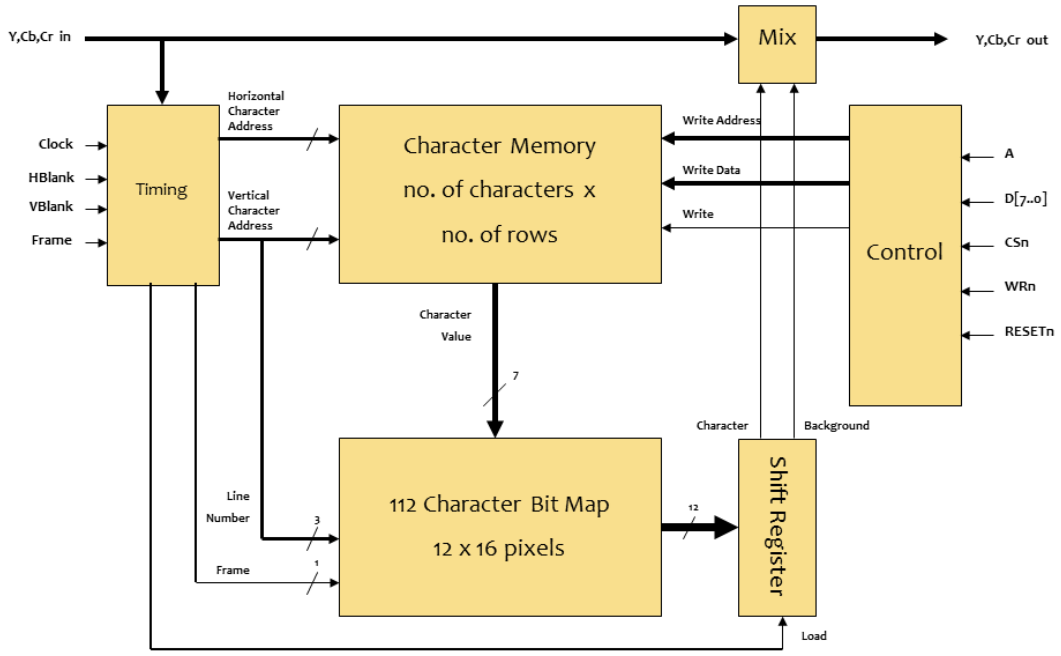

Figure 3 PT57 Block diagram

The high to low transition of the HBlank input triggers the left margin counter which runs until its count matches that programmed in registers $00 and $01. This sets the left margin before the start of the character display. Each character is 12 x 16 pixels in size. Once the active character display starts, one counter counts from 0 to 11 to address the column of the character selected, while another counts the no of characters and addresses the programmable character memory to select the character to display. The width of the active character area is set by the parameter .no_Hchar. For example, if we wish to have a centred display of 64 characters for a 720p video standard: 720p standard has 1280 active pixels/line. 64 characters will be 64 x 12 pixels wide = 768 pixels. 1280 – 768 = 512 pixels. So we set parameter .no_Hchar to 64 (and parameter H_addr_width to ceil(log2[.no_Hchar]) = 6) and program Left margin registers to 512/2 = 256 pixels.

Similarly, for the vertical positioning, the high to low transition of the VBlank input starts the top margin counter which counts horizontal lines until they match the programmed values in registers $02 and $03. The active character area is then the parameter .no_rows x 16 (each character is 16 lines high). Parameter .V_addr_height must also be set to ceil(log2[.no_rows]). The lower 4 address bits address the character ROM row and the upper bits select the programmable character memory to select the programmed character.

The Standard[3:0] select the video standard as shown in Table 3. If the input is an interlaced standard, control register bit [0] must be set to '1' and the 'Frame' signal input (indicating odd or even field) modifies the addressing to the character ROM. For progressive video standards, the 'Frame' input should be set to '0' and control register bit [0] is set to '0'.

Standard[3:0]	Format	Pixels/line	Line frequency
0000	720p/25Hz	3960	18.75kHz
0001	720p/30Hz	3300	22.5kHz
0010	720p/50Hz	1980	37.500kHz
0011	720p/59.94Hz[1]	1650	44.955kHz

0100	720p/60Hz	1650	45.000kHz
0101	1080p/24Hz	2750	27.0kHz
0110	1080p/25Hz	2640	28.125kHz
0111	1080p/29.97Hz[1]	2200	33.716kHz
1000	1080p/30Hz	2200	33.750kHz
1001	1080i/50Hz	2640	28.125kHz
1010	1080i/59.94Hz[1]	2200	33.716kHz
1011	1080i/60Hz	2200	33.750kHz

[1] **Input clock is 74.1758242MHz (else 74.25MHz).**

Table 3 Supported video standards.

The addressing of the character is shown graphically in Figure 4.

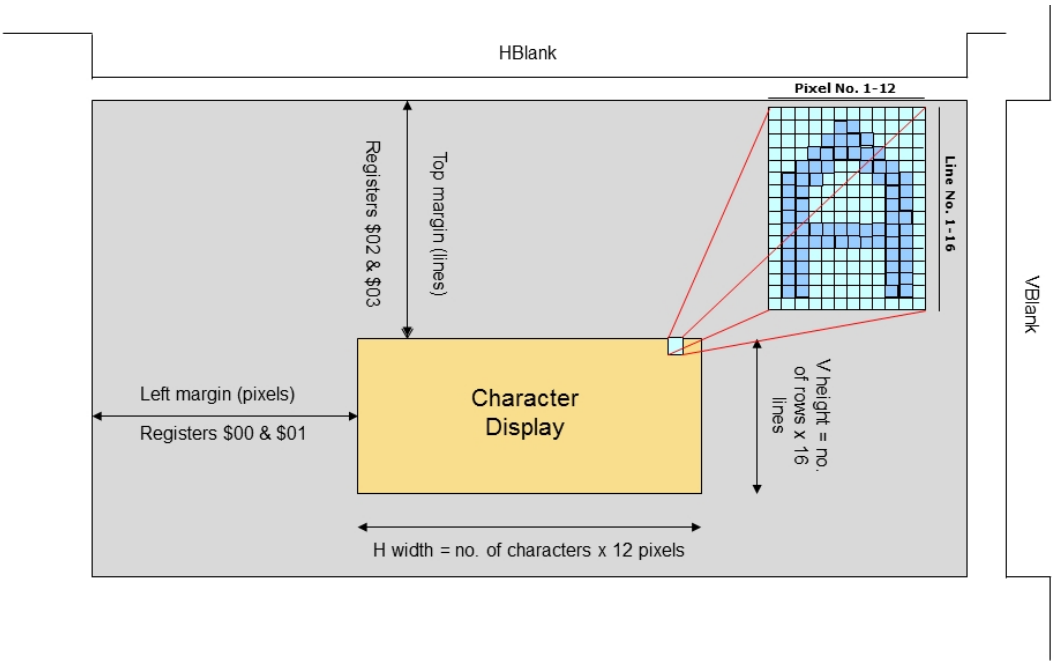

Figure 4 Programming the character position.

Each line of the character display runs to the value $2^{(H\_addr\_width)}$ -1 even if the character value is less than that: for example, if you set the no. characters/line to 50, the character counter will run to 2^6 - 1 = 63 (although only 50 characters will be displayed). This means the address of the first character on the next row will be 64 and not 51.

To write to the character memory CharGen_CSn must be taken low and then PT57_WRn is taken low. On the rising edge of PT57_WRn the character is written into memory. This process takes 4 x Clock periods, or 54ns at a 74.25MHz clock. Both the Din[7:0] and A[ram_addr_width-1:0] must remain valid during this period (see Figure 5).

Writing to memory is not interleaved and can cause a minor 'flash' on the screen. To avoid this writing should be performed during the blanking intervals.

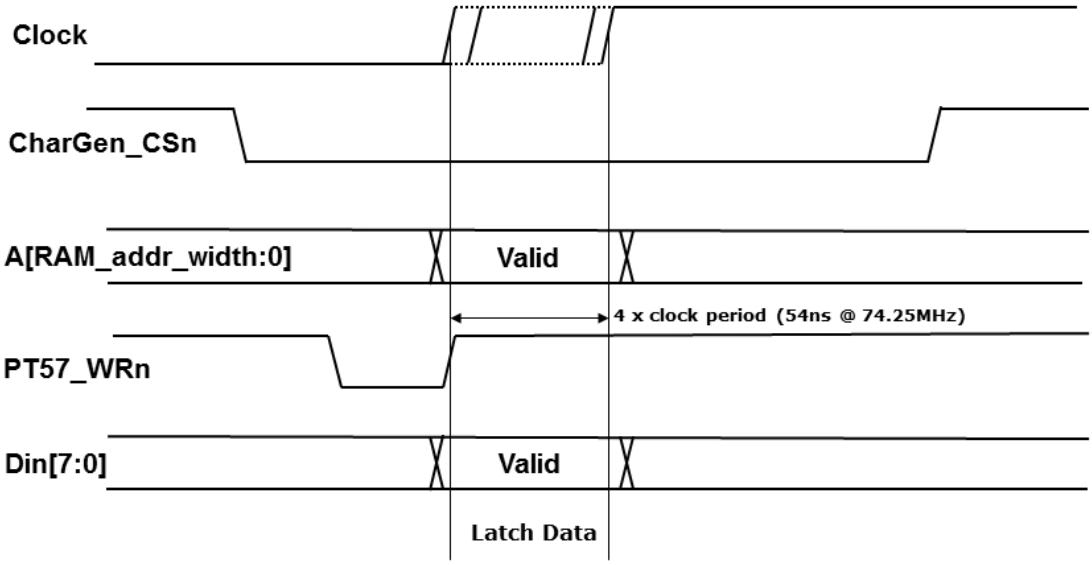

Figure 5 Character RAM writing.

The output of the character RAM selects one of 112 characters; each character is 12 pixels wide (13.5ns/pixel) by 16 lines high. The characters available are shown in Table 4.

Hex Code		Hex Code		Hex Code		Hex Code	
$00	[1]	$10	F	$20	V	$30	l
$01	0	$11	G	$21	W	$31	m
$02	1	$12	H	$22	X	$32	n
$03	2	$13	I	$23	Y	$33	o
$04	3	$14	J	$24	Z	$34	p
$05	4	$15	K	$25	a	$35	q
$06	5	$16	L	$26	b	$36	r
$07	6	$17	M	$27	c	$37	s
$08	7	$18	N	$28	d	$38	t
$09	8	$19	O	$29	e	$39	u
$0A	9	$1A	P	$2A	f	$3A	v
$0B	A	$1B	Q	$2B	g	$3B	w
$0C	B	$1C	R	$2C	h	$3C	x

Hex Code		Hex Code		Hex Code		Hex Code	
$0D		$1D		$2D		$3D	
$0E		$1E		$2E		$3E	
$0F		$1F		$2F		$3F[Note 2]	
$40		$50		$60			
$41		$51		$61			
$42		$52		$62			
$43		$53		$63			
$44		$54		$64			
$45		$55		$65			
$46		$56		$66			
$47		$57		$67			
$48		$58		$68			
$49		$59		$69			
$4A		$5A		$6A			
$4B		$5B		$6B			
$4C		$5C		$6C			
$4D		$5D		$6D			
$4E		$5E		$6E			
$4F		$5F		$6F			
Note 1	Value ‘0’ displays nothing. All other values automatically turn on the background if it is enabled in the control register 1, bit 1.						
Note 2	Value ‘$3F’ is a space. i.e. it displays background only, (if enabled), but no character.						

Table 4 Pre-programmed Characters

The colour of each character and the background is programmable (registers $05-$0A). Example values are shown in Table 5.

Colour	Y value (Register $05 & $08)	Cb value (Register $06 & $09)	Cr value (Register $07 & $0A)	
White	235	128	128	
Grey	180	128	128	
Yellow	161	44	142	
Cyan	131	156	44	
Green	112	72	58	
Magenta	83	183	198	
Red	65	100	212	
Blue	34	212	114	
Black	16	128	128	

Table 5 Overlay and Background value settings

The character is mixed with the input video and the resulting video output is valid on the rising edge of 'Clock'. The PT57 has a single clock pipeline delay.

4. Register Interface

Figure 6 shows the timing diagram for the register interface; it is a conventional microprocessor interface.

Each register is selected via an 4 bit address bus. Writes to unused register locations are ignored.

To write to the selected register the PT57_CSn (chip select) input must be asserted low. Whilst this is low the PT57_WRn must be taken low. On the rising edge of PT57_WRn the register is written. A[3:0] and Din[7:0] must be valid during this rising edge.

There is no read back of the register values (registers are write only).

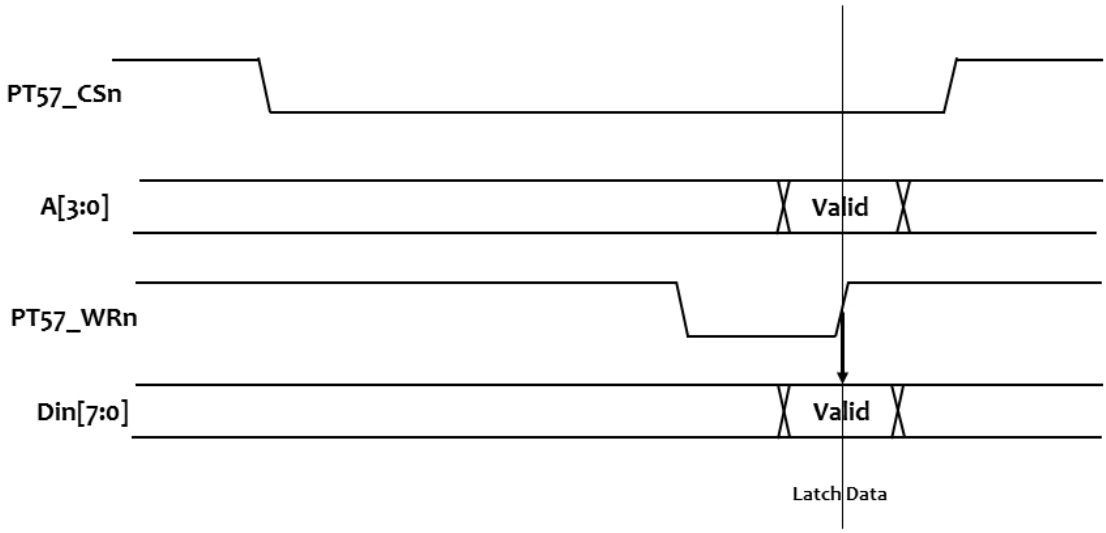

Figure 6 Register timing.

5. Register Descriptions

Table 6 lists the control registers. All of the registers are 8 bits wide. There is no read back of the register values.

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

Asserting the RESETn input sets the PT57 to default values.

Register Offset	Register Name	R/W	Bit Value	Description
$00	**Left margin LSB**	W	7:0	Sets the value of the left margin (see Figure 4). 12 bit value: ({Left margin MSB[3:0],Left margin LSB[7:0]}) x 1/Clock. Default value = 256_{10}
$01	**Left margin MSB**	W	3:0	
$02	**Top margin LSB**	W	7:0	Sets the value of the top margin (see Figure 4). 11 bit value: ({Top margin MSB[2:0],Top margin LSB[7:0]}) x no. horizontal lines. Default value = 592_{10}.
$03	**Top margin MSB**	W	2:0	
$04	**Control 1**	W		Control Register
			7	Enables the character display if set to '1', else the overlay is disabled. Default value = '0'.
			6	Enables the background if set to '1', else the background is disabled. Default value = '0'.
			5	If set to '1' the luma background is ¼ amplitude video, else the luma background is set by register $05. Default value = '0'.
			4-1	Not used.
			0	If set the '1' the input video standard is interlaced, else the input standard is progressive. Default value = '0'.
$05	**Y Overlay value**	W		Sets the level of the overlay character Y (luma) value. **Must be limited to between values 16_{10} and 235_{10}. See Table 5.** Default value = 161_{10} (yellow).
$06	**Cb Overlay value**	W		Sets the level of the overlay character Cb (chroma) value. **Must be limited to between values 16_{10} and 235_{10}. See Table 5.** Default value = 44_{10} (yellow).
$07	**Cr Overlay value**	W		Sets the level of the overlay character Cr (chroma) value. **Must be limited to between values 16_{10} and 235_{10}. See Table 5.** Default value = 142_{10} (yellow).
$08	**Y Background value**	W		Sets the level of the background Y (luma) value (register $04 bit 5 = '0'). **Must be limited to between values 16_{10} and 235_{10}. See Table 5.** Default value = 16_{10} (black).
$09	**Cb Background value**	W		Sets the level of the background Cb (chroma) value). **Must be limited to between values 16_{10} and 235_{10}. See Table 5.** Default value = 128_{10} (black).
$0A	**Cr Background value**	W		Sets the level of the background Cr (chroma) value). **Must be limited to between values 16_{10} and 235_{10}. See Table 5.** Default value = 128_{10} (black).

Table 6 Register descriptions.