

PT2

User Manual

BT656 Video

Character Generator

Revision 0.2
20th September 2008

Contents

Contents 2
Tables..... 2
Figures..... 2
Introduction..... 3
Signal Interconnections..... 4
Technical Overview..... 6
Register interface 9
Register descriptions..... 10
Display Memory Interface..... 11
Character Memory Interface 12

Tables

Table 1 FPGA resource requirements..... 3
Table 2 Input/Output signals..... 5
Table 3 Pre-programmed characters..... 8
Table 4 Register descriptions 10

Figures

Figure 1 PT2 Symbol Block Diagram 4
Figure 2 PT2 Block diagram 6
Figure 3 Character mapping 7
Figure 4 PT2 Screen Capture..... 8
Figure 5 PT2 Register Timing diagram 9
Figure 6 PT2 Memory Timing Diagram 11
Figure 7 Character Description 12
Figure 8 Character Generator memory 14

Introduction

PT2 is a video character/text overlay IP core for FPGAs; PT2 is suitable for the display of closed caption or subtitles, user menus, status or error information, time-code or channel identifiers.

The core accepts a compliant BT656 video stream and associated 27MHz clock from which it extracts the timing information. The character overlay is 60x32 characters - (60x30 for 525 lines) - and each character is 12x16 pixels. The character memory can store up to 128 characters and is pre-programmed with the characters, 0-9, A-Z, a-z, space and various punctuation marks.

Additionally the character may be displayed a background. The luminance level of the both the character and the background are programmable.

Both the character memory and the character bit map memories are user programmable. The memories and the control registers are written to and read from using a conventional 8 bit wide microprocessor interface.

The intellectual property block is provided as an encrypted 'black box' design for incorporation into an Altera FPGA, either for stand-alone operation or for integration with other blocks. The resources required for an Altera FPGA are shown in Table 1.

Logic Elements	Memory Bits	M4K blocks	9x9 Multipliers	18x18 multipliers
151	38912	10	0	0

Table 1 FPGA resource requirements

Signal Interconnections

The PT2 signal interconnect diagram is shown in Figure 1.

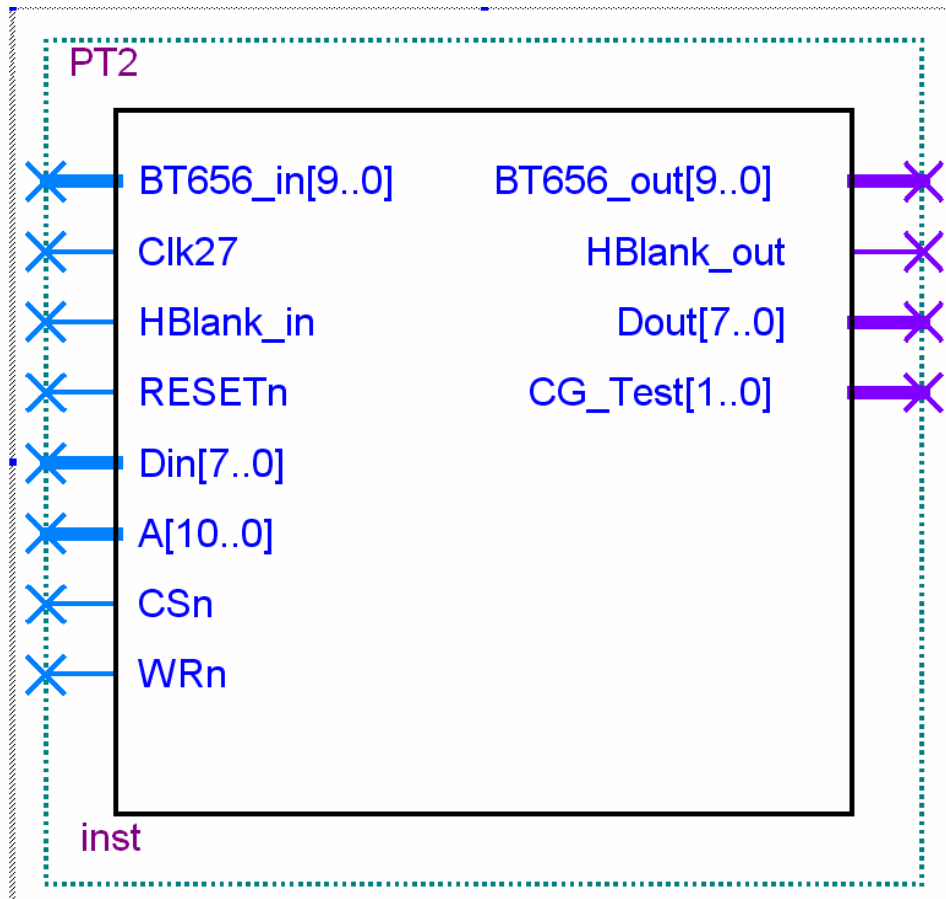


Figure 1 PT2 Symbol Block Diagram

The signal descriptions are shown in Table 2, below.

Signal	Description
BT656_in[9..0]	BT656 video input, synchronous with Clk27. If only eight bits are used the bottom two bits should be tied to ground.
Clk27	27MHz clock input. The input data, BT656_in[9..0] is latched on the rising edge of this clock and the output data D1_out[9..0] is valid on the rising edge of this clock.
HBlank_in	Horizontal blanking input. If unused it should be tied to ground.
RESETn	Active low reset signal. Asserting this input sets all the control registers to their default value.
Din[7..0]	Control data input bus.
A[10..0]	Control address bus input used to select the

	control register/memory location to be written to/read from.
CSn	Control chip select input, active low. Used in combination with the WRn input to control writing to the control registers.
WRn	Active low write enable input. Used in combination with the CSn input to control writing to the control registers.
BT656_out[9..0]	The 10-bit, BT656 formatted output video, synchronous with the input 27MHz clock, Clk27.
HBlank_out	Horizontal blanking output, a delayed version of HBlank_in.
Dout[7..0]	Control output data bus. Outputs the control/status register data selected by the A[9..0] bus.
CG_Test[1..0]	Used in combination with the test control register to select internal test signals.

Table 2 Input/Output signals

The Altera include file is shown below:

```
FUNCTION PT2 (BT656_in[9..0], Clk27, HBlank_in, RESETn, Din[7..0],
A[10..0], CSn, WRn)
RETURNS (BT656_out[9..0], HBlank_out, Dout[7..0], CG_Test[1..0]);
```

An example of instantiating PT2 is shown below:

```
-- *****
-- Character Generator
(BT656_out[9..0], HBlank_out, CG_Dout[7..0],CG_Test[1..0]) =
PT2(Pattern_out[9..0], XTAL_clk, HBlank, RESETn, RAM_data_in[7..0],
RAM_A[10..0], CG_CSn, read_write);
```

Technical Overview

A simplified block diagram of the PT2 character generator is shown in Figure 2.

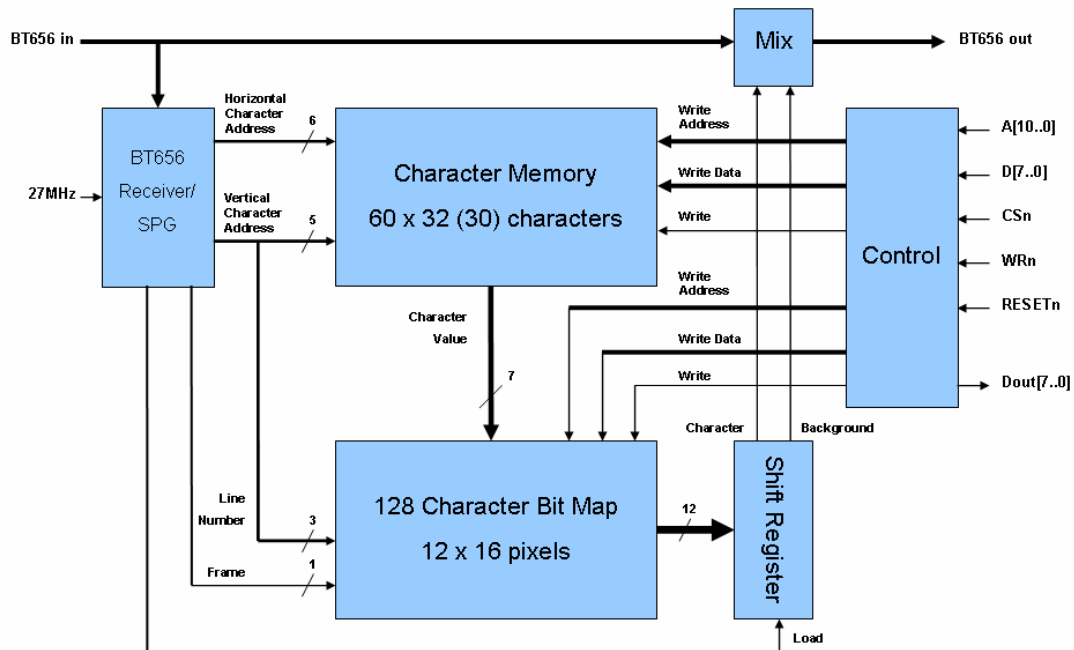


Figure 2 PT2 Block diagram

The sync pulse generator detects the embedded TRS signals in the BT656 stream and generates horizontal and vertical addresses for the character memory. The character memory is written to via the microprocessor control port. Each horizontal row of characters is 64 bytes long, of which the first 60 are displayed, (the 'invisible' character bytes are used for control registers). There are 32 rows of characters for 625 line formats and 30 rows for 525 line formats.

The addressing of the character is shown graphically in Figure 3. The character display is linearly memory mapped. Each horizontal row has 64 addressable characters of which only 60, (address 0-59), are displayed. The rightmost characters, address 60-63, of the upper addresses are used for the control registers. Address lines A[5..0] are used to select the horizontal character.

Each row from the top to the bottom of the image is contiguous, so the first (top) row starts at address 0 and finishes at address 63 (with addresses 60-63 not displayed). The second row starts at

address 64 and ends at address 127 with addresses 124-127 not displayed. Address lines A[10..6] select one of the 32 rows, (30 for 525 lines standards). 625 line standards have 576 active (displayed) lines. However 32 rows of characters, (each character being 16 lines high), occupy 512 lines leaving 64 lines or 4 rows that cannot be written to. The characters are arranged so that there are 2 rows at the top and bottom of the screen that cannot be written to.

Note the character display is write only and it cannot be read from, (reading will return value \$00).

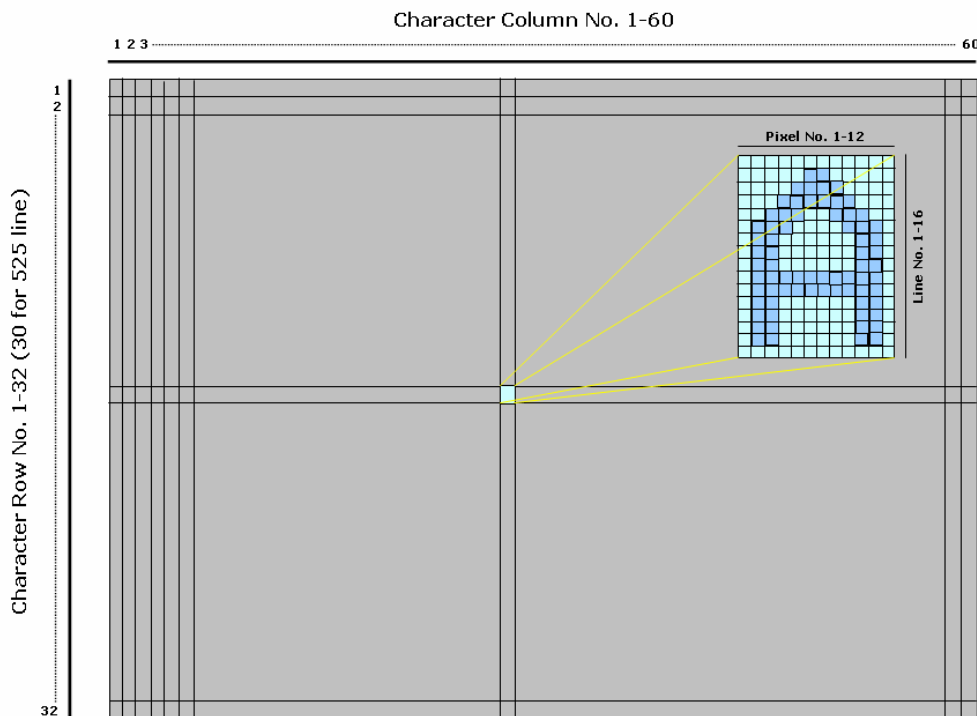


Figure 3 Character mapping

The output of the character memory selects one of 128 characters; each character is 12 pixels wide (74ns/pixel) by 16 lines high. A number of the characters are pre-programmed, see Table 3.

Value	Character	Value	Character	Value	Character	Value	Character
\$000	Note 1	\$020	V	\$040	-	\$060	
\$001	0	\$021	W	\$041	:	\$061	
\$002	1	\$022	X	\$042	.	\$062	
\$003	2	\$023	Y	\$043	/	\$063	
\$004	3	\$024	Z	\$044	'	\$064	
\$005	4	\$025	a	\$045	?	\$065	
\$006	5	\$026	b	\$046	=	\$066	
\$007	6	\$027	c	\$047		\$067	
\$008	7	\$028	d	\$048		\$068	
\$009	8	\$029	e	\$049		\$069	
\$00A	9	\$02A	f	\$04A		\$06A	

SingMai Electronics

Value	Character	Value	Character	Value	Character	Value	Character
\$00B	A	\$02B	g	\$04B		\$06B	
\$00C	B	\$02C	h	\$04C		\$06C	
\$00D	C	\$02D	i	\$04D		\$06D	
\$00E	D	\$02E	j	\$04E		\$06E	
\$00F	E	\$02F	k	\$04F		\$06F	
\$010	F	\$030	l	\$050		\$070	
\$011	G	\$031	m	\$051		\$071	
\$012	H	\$032	n	\$052		\$072	
\$013	I	\$033	o	\$053		\$073	
\$014	J	\$034	p	\$054		\$074	
\$015	K	\$035	q	\$055		\$075	
\$016	L	\$036	r	\$056		\$076	
\$017	M	\$037	s	\$057		\$077	
\$018	N	\$038	t	\$058		\$078	
\$019	O	\$039	u	\$059		\$079	
\$01A	P	\$03A	v	\$05A		\$07A	
\$01B	Q	\$03B	w	\$05B		\$07B	
\$01C	R	\$03C	x	\$05C		\$07C	
\$01D	S	\$03D	y	\$05D		\$07D	
\$01E	T	\$03E	z	\$05E		\$07E	
\$01F	U	\$03F		\$05F		\$07F	Note 2

Table 3 Pre-programmed characters

Note 1: Value '0' displays nothing. All other values automatically turn on the background if it enable in the control register 1, bit 1.

Note 2: Value '\$7F' is a space. i.e. it displays background only, (if enabled), but no character.

The character bit maps can also be written to by the user allowing custom characters to be created.

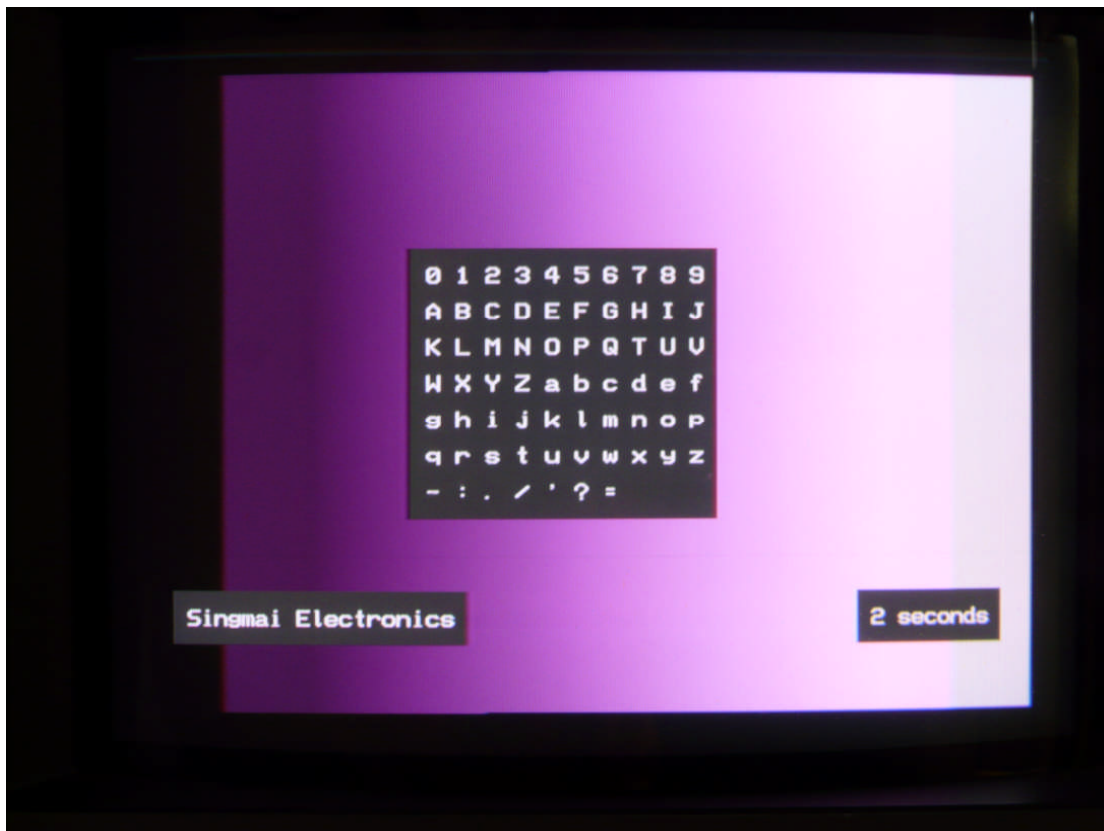


Figure 4 PT2 Screen Capture

Register interface

Figure 5 shows the timing diagram for the register interface. It is effectively a conventional microprocessor interface. The registers locations are shared with unused RAM locations and are selected using the same 11 bit address bus. Writes to unused register locations are ignored.

To write to the selected register the CSn (chip select) input must be asserted low. Whilst this is low the WRn must be strobed low and then high. Writing occurs on the low to high transition of this pulse, and therefore the data to be written must be stable 10ns before and after the transition. Note that writing will not occur correctly if the CSn or address inputs are not valid during that time.

The address input also selects the register data that is presented on the Dout[7..0] bus. This output is independent of the other control signals or the Clk27 clock.

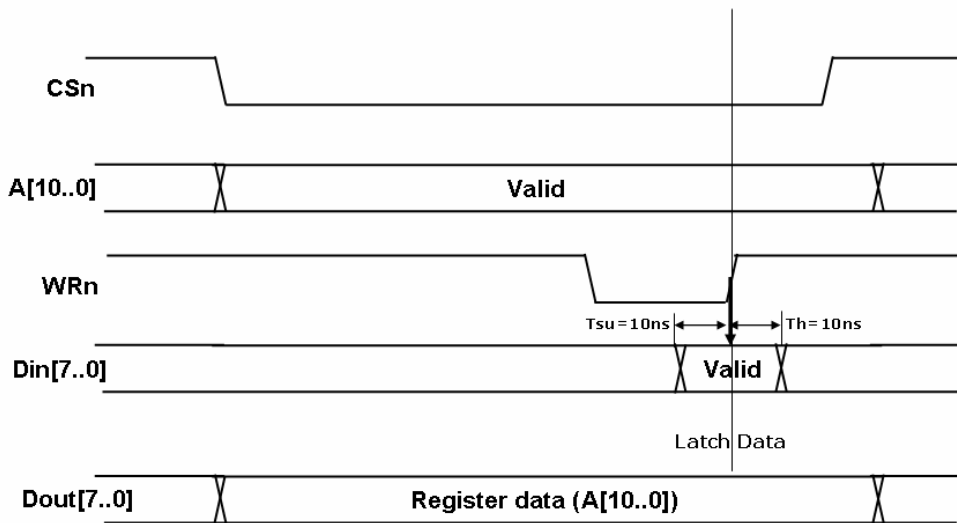


Figure 5 PT2 Register Timing diagram

Register descriptions

The following table lists all of the control and status registers. All of the registers are 8 bit wide although some are concatenated together to create longer words. Asserting the RESETn input sets all the registers to their default values. Unused bits read back as '0's.

Offset	Register Name	R/W	Bit Value	Default Value	Description
Pattern Memory					
\$000-\$7FF	Character Memory	W		\$00	See Memory interfacing for detailed description. Some of the non displayed character locations are used for the register controls.
Control and Status					
\$7FF	Version No.	R		\$01	IP version number.
\$7FE	Control 1	R/W		0	
	625_525n		0	1	Selects the line standard to be 625 = '1' or 525 = '0'.
	Character_enable		1	0	Enables the character display if '1'.
	Background_enable		2	0	Enables the background if '1'.
\$7FD	Character value	R/W		\$F8	The luminance value of the character. \$00 is black and \$FF is white, (\$80 is mid grey).
\$7FC	Background value	R/W		\$2A	The luminance value of the background. \$00 is black and \$FF is white, (\$80 is mid grey).
Character Memory control					
\$7BF	Char_ROM_data MSB	R/W		\$0	Bits 3..0 are the top 4 bits of the data to be written to the character memory. Unused bits read back as '0'.
\$7BE	Char_ROM_data LSB	R/W		\$00	Bits 7..0 are the bottom 8 bits of the data to be written to the character memory.
\$7BD	Char_ROM_addr MSB	R/W		\$0	Bits 2..0 are the top 3 bits of the address for the character memory. Unused bits read back as '0'.
\$7BC	Char_ROM_addr LSB	R/W		\$00	Bits 7..0 are the bottom 8 bits of the address for the character memory.
\$77F	Char_ROM_write	W		\$0	Writing a '1' to bit 0 initiates a write to the character memory at the address selected by Char_ROM_addr[10..0] with the data Char_ROM_data[11..0]. The bit is automatically cleared after the write cycle has been performed.

Table 4 Register descriptions

Display Memory Interface

Figure 6 shows the timing diagram for the memory interface.

Writing to the memory is synchronous with the 27MHz video input clock. CSn and WRn should be asserted low and the address and data input also be valid. On the next rising edge of the 27MHz clock the control signals and data will be latched into the memory. To ensure an occurrence of the rising edge of the clock these signals should be asserted and valid for a minimum of 74ns.

The address input also selects the register data that is presented on the Dout[7..0] bus. This output is independent of the other control signals or the Clk27 clock.

The character display memory is write only; reading from the memory will return a null value, (\$00). Also some of the memory locations that are unused are used for register locations, (see Register descriptions).

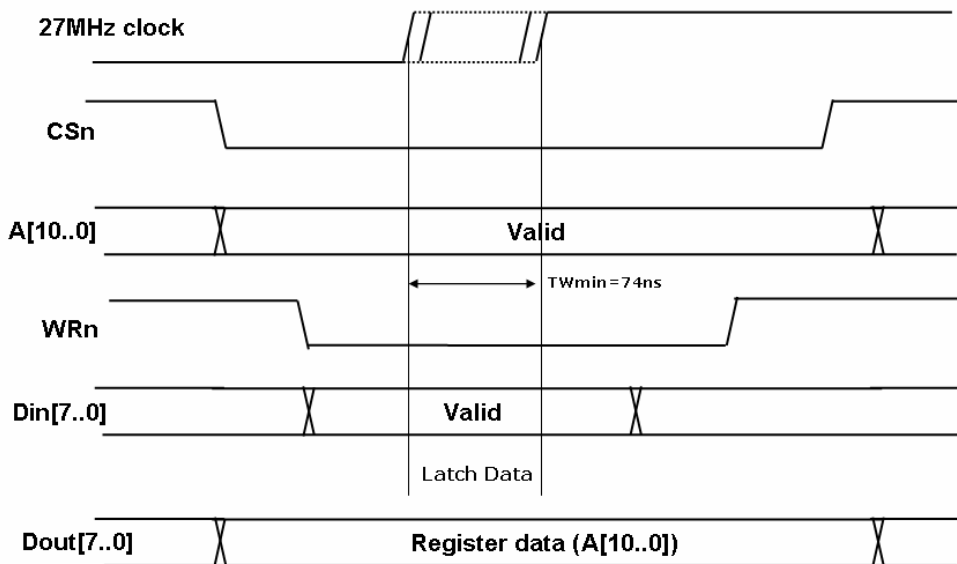


Figure 6 PT2 Memory Timing Diagram

Character Memory Interface

At switch on the character memory is preset with a standard set of characters, see Table 3. However all of these locations and also the spare ones may be programmed with user characters or symbols.

Each of the 128 available characters is 12 pixels wide by 16 lines high. The character memory is arranged as 12 bits deep representing the 12 horizontal pixels and 128 x 16 = 2048 words deep. Each location in the memory therefore contains, for each of the 128 characters, the 12 horizontal pixels for any particular vertical line of that character. For example, in Figure 7, we show the character 'A' which is address '\$0b', see Table 3. The character address is the upper 7 bits of the character memory, A[10..4], so character 'A' is at address \$0bx. The lower 4 bits of the memory select which line of the character is displayed, Line '0', the top line of the character, is address A[3..0] = '0' (shown in grey in Figure 6). Address \$0b1 selects the second line of the character, shown in green, and address \$0bF selects the bottom line of the character, shown in blue.

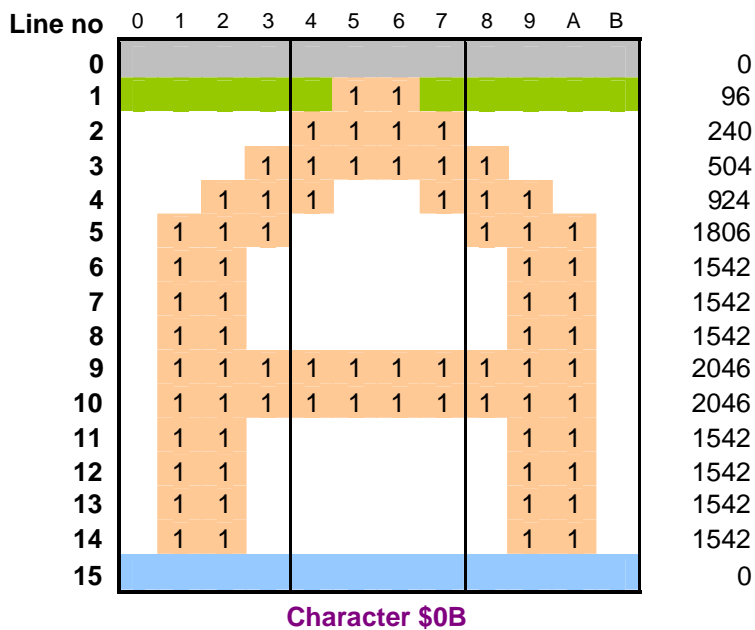


Figure 7 Character Description

Each 12 bit word of the character memory represents the data for the twelve horizontal character pixels for that selected vertical line. A '1' in any bit location of the word displays the character at the selected character value; a '0' displays just the background, if enabled.

The left most pixel of the character is the most significant bit of the word and the right most pixel is the least significant. The word can therefore be constructed from the equation:

$$P0*2^{11} + P1*2^{10} + P2*2^9 + P3*2^8 + P4*2^7 + P5*2^6 + P6*2^5 + P7*2^4 + P8*2^3 + P9*2^2 + P10*2^1 + P11*2^0$$

If we look at line 1 of Figure 7, highlighted in green, there are two 'active' pixels at locations P5 and P6 so we can insert those values into the equation:

$$0*2^{11} + 0*2^{10} + 0*2^9 + 0*2^8 + 0*2^7 + 1*2^6 + 1*2^5 + 0*2^4 + 0*2^3 + 0*2^2 + 0*2^1 + 0*2^0 = (32 + 64) = 96.$$

96 is therefore the value that should be written into location \$0b1 for character 'A'.

A portion of the pre-programmed character memory is shown in Figure 8. We can see for address \$0b1 the value 96 is written, the other vertical line values for character 'A' can be seen in Figure 7 and 8

An Excel file is provided with the IP core which allows you to easily calculate the values for each character; just write a '1' in the Excel location where you want a pixel highlighted and the resulting word value is calculated. In the file the pixel is also shaded to more easily see it.

Writing to the character memory is done by register indexing. First the register address needs to be loaded into memory, then the data, and then a write command issued. Registers \$7BF (MSB bits 11..8), and \$7BE (LSB, bits 7..0), hold the data to be written. So if we wanted to write a solid line of pixels we would write \$0F to register \$7BF and \$FF to register \$7BE. We then need to store the location to be written to. If we wanted to write a solid line on the top of the character 'A' at address \$0b0, we would write \$0 to register \$7BD (MSB bits 10..8), and \$b0 to register \$7BC (MSB, bits 7..0). We then initiate a write by writing a '1' to register \$77F, bit 0. This will write data \$FFF to location \$0b0. After the write cycle is complete bit 0 of register \$77F is automatically cleared. If we then wanted to write another solid line below the one just written we would then write \$b1 to register \$7BC and write a '1' to register \$77F bit 0 to initiate another write cycle.

SingMai Electronics

Addr	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+a	+b	+c	+d	+e	+f
000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010	0	504	1020	1806	1550	1566	1590	1638	1734	1926	1798	1542	1806	1020	504	0
020	0	96	96	480	480	96	96	96	96	96	96	96	96	504	504	0
030	0	504	1020	1806	1542	6	14	508	1016	1792	1536	1536	1536	2046	2046	0
040	0	504	1020	1806	1542	6	14	124	124	14	6	1542	1806	1020	504	0
050	0	120	248	472	920	1816	1560	1560	1560	2046	2046	24	24	24	24	0
060	0	2046	2046	1536	1536	2040	2044	14	6	6	6	1542	1806	1020	496	0
070	0	504	1020	1806	1542	1536	1536	2040	2044	1550	1542	1542	1804	1020	504	0
080	0	2046	2046	6	12	24	48	96	96	96	96	96	96	96	96	0
090	0	504	1020	1806	1542	1542	1806	1020	1020	1542	1542	1542	1806	1020	504	0
0a0	0	504	1020	1806	1542	1542	1798	1022	510	6	6	1542	1806	1020	504	0
0b0	0	96	240	504	924	1806	1542	1542	1542	2046	2046	1542	1542	1542	1542	0
0c0	0	2040	2044	1550	1542	1542	1550	2044	2044	1550	1542	1542	1550	2044	2040	0
0d0	0	504	1020	1806	1536	1536	1536	1536	1536	1536	1536	1536	1806	1020	504	0
0e0	0	2040	2044	1550	1542	1542	1542	1542	1542	1542	1542	1542	1550	2044	2040	0
0f0	0	2046	2046	1536	1536	1536	1536	2040	2040	1536	1536	1536	1536	2046	2046	0
100	0	2046	2046	1536	1536	1536	1536	2040	2040	1536	1536	1536	1536	1536	1536	0
110	0	504	1020	1806	1542	1536	1536	1598	1598	1542	1542	1542	1806	1020	504	0
120	0	1542	1542	1542	1542	1542	1542	2046	2046	1542	1542	1542	1542	1542	1542	0
130	0	504	504	96	96	96	96	96	96	96	96	96	96	504	504	0
140	0	126	126	24	24	24	24	24	24	24	24	1560	1848	1008	480	0
150	0	1542	1550	1564	1592	1648	2016	1984	1984	1760	1648	1592	1564	1550	1542	0
160	0	1536	1536	1536	1536	1536	1536	1536	1536	1536	1536	1536	1536	2046	2046	0
170	0	1542	1806	1950	2046	1782	1638	1542	1542	1542	1542	1542	1542	1542	1542	0
180	0	1542	1542	1542	1798	1926	1990	1766	1654	1598	1566	1550	1542	1542	1542	0
190	0	504	1020	1806	1542	1542	1542	1542	1542	1542	1542	1542	1806	1020	504	0
1a0	0	2040	2044	1550	1542	1542	1550	2044	2040	1536	1536	1536	1536	1536	1536	0
1b0	0	504	1020	1806	1542	1542	1542	1542	1542	1638	1654	1598	1820	1022	486	0
1c0	0	2040	2044	1550	1542	1542	1550	2044	2040	1760	1648	1592	1564	1550	1542	0
1d0	0	510	1022	1792	1536	1536	1792	1016	508	14	6	6	14	2044	2040	0
1e0	0	2046	2046	96	96	96	96	96	96	96	96	96	96	96	96	0
1f0	0	1542	1542	1542	1542	1542	1542	1542	1542	1542	1542	1542	1806	1020	504	0
200	0	1542	1542	1542	1542	1542	1542	1542	1542	1542	1806	924	504	240	96	0
210	0	1542	1542	1542	1542	1542	1542	1542	1638	1782	2046	1950	1806	1542	1542	0
220	0	1542	1542	1542	1806	924	504	240	240	504	924	1806	1542	1542	1542	0
230	0	1542	1542	1542	1542	1806	924	504	240	96	96	96	96	96	96	0
240	0	2046	2046	6	14	28	56	112	224	448	896	1792	1536	2046	2046	0
250	0	0	0	0	0	496	1016	796	12	508	1020	1804	1548	2046	1014	0
260	0	1536	1536	1536	1536	1784	2044	1806	1542	1542	1542	1542	1550	2044	2040	0
270	0	0	0	0	0	504	1020	1806	1536	1536	1536	1536	1806	1020	504	0
280	0	12	12	12	12	508	1020	780	1548	1548	1548	1548	780	1022	510	0
290	0	0	0	0	0	504	1020	1806	1542	2046	2046	1536	1806	1020	504	0
2a0	0	120	252	204	192	192	1008	1008	192	192	192	192	192	192	192	0
2b0	0	0	0	0	0	502	1022	1806	1542	1806	1022	502	6	1806	1020	504
2c0	0	1536	1536	1536	1536	1656	1788	1934	1798	1542	1542	1542	1542	1542	1542	0
2d0	0	96	96	0	0	224	96	96	96	96	96	96	96	96	504	0
2e0	0	24	24	0	0	56	24	24	24	24	24	24	792	952	504	240
2f0	0	1536	1536	1536	1536	1542	1550	1564	1592	1648	1776	2040	1948	1550	1542	0
300	0	224	96	96	96	96	96	96	96	96	96	96	104	120	48	0
310	0	0	0	0	0	1756	1900	1638	1638	1638	1638	1638	1638	1638	1638	0
320	0	0	0	0	0	1784	2044	1806	1798	1542	1542	1542	1542	1542	1542	0
330	0	0	0	0	0	504	1020	1806	1542	1542	1542	1542	1806	1020	504	0
340	0	0	0	0	0	1784	2044	1806	1542	1542	1806	2044	1784	1536	1536	1536
350	0	0	0	0	0	502	1022	1806	1542	1542	1806	1022	502	6	6	6
360	0	0	0	0	0	1784	2044	1934	1798	1536	1536	1536	1536	1536	1536	0
370	0	0	0	0	0	504	1020	1806	1792	1016	1020	14	1806	1020	504	0
380	0	96	96	96	96	504	504	96	96	96	96	96	96	120	56	0
390	0	0	0	0	0	1542	1542	1542	1542	1542	1542	1542	1806	1022	502	0
3a0	0	0	0	0	0	1542	1542	1542	1542	1542	1806	924	504	240	96	0

Figure 8 Character Generator memory