

# **FSA506 Application Note**

---

## **a-Si TFT LCD Controller 640xRGBx240 Resolution 262K color**

Vresion 1.0  
Date:2008/01/07

# FSA506 Application Note

---

<b>1</b>	<b>RECORD OF REVISION .....</b>	<b>3</b>
<b>2</b>	<b>FEATURES .....</b>	<b>4</b>
<b>3</b>	<b>BLOCK DIAGRAM .....</b>	<b>5</b>
<b>4</b>	<b>PIN DESCRIPTION.....</b>	<b>5</b>
4.1	PIN ASSIGNMENT .....	5
4.1.1	<i>Power Supply Pins</i> .....	6
4.1.2	<i>Output Pins</i> .....	7
4.1.3	<i>Input Pins</i> .....	8
<b>5</b>	<b>APPLICATION CIRCUIT.....</b>	<b>9</b>
<b>6</b>	<b>JUMPER SETTING .....</b>	<b>11</b>
<b>7</b>	<b>INTERFACE PROTOCOL .....</b>	<b>12</b>
7.1	18BIT-80/68-WRITE TO COMMAND REGISTER.....	12
7.2	18BIT-80/68-WRITE TO DISPLAY RAM .....	13
7.3	16BIT-80/68- WRITE TO COMMAND REGISTER.....	13
7.4	16BIT-80/68-WRITE TO DISPLAY RAM .....	14
7.5	9BIT-80/68- WRITE TO COMMAND REGISTER.....	15
7.6	9BIT-80/68-WRITE TO DISPLAY RAM .....	15
7.7	8BIT-80/68- WRITE TO COMMAND REGISTER.....	16
7.8	8BIT-80/68-WRITE TO DISPLAY RAM .....	17
7.9	DATA TRANSFER ORDER SETTING .....	18
<b>8</b>	<b>REGISTER DEPICTION .....</b>	<b>19</b>
<b>9</b>	<b>REFERENCE INITIAL CODE : .....</b>	<b>28</b>
9.1	8BIT-80 INTERFACE MODE , 65K COLOR , PANEL:320XRGBx240 , LANDSCAPE .....	28
9.2	8BIT-80 INTERFACE MODE , 65K COLOR , PANEL:320XRGBx240 , PORTRAIT.....	34

# FSA506 Application Note

---

## 1 RECORD OF REVISION

Revision Date	Page	Contents	Editor
2008/1/7	-	New Release	

# FSA506 Application Note

---

## 2 Features

FSA506 is a CPU interface based TFT LCD controller. It can support panel resolution up to 640x240 pixels with 262144 colors depth. User can send either a full screen picture or a partial image by controlling the MPU with popular microprocessor interface :

- 18 bit 8080-series MPU
- 16 bit 8080-Series MPU
- 9 bit 8080-Series MPU
- 8 bit 8080-Series MPU
- 18 bit 6800-Series MPU
- 16 bit 6800-Series MPU
- 9 bit 6800-Series MPU
- 8 bit 6800-Series MPU

The integrated on-chip functions include:

- Supported panel resolution
- - Max. 153600 pixels ( H x V ) resolution
- Built-in separated RGB SRAM with 18-bits addressing space
- Both delta and stripe type panel supported
- Build in Images rotate and shift function
- Arbitrary display memory address access
- Supply voltage: 3.3v
- 100 pin LQFP package



# FSA506 Application Note

---

## 4.1.1 Power Supply Pins

Pin-No	Symbol	I/O	Description	Remark
1	R_V18	P	Regulator 1.8V output	
2	VCCA_H	P	Regulator 3.3V Power in	
3	GNDA_REG	P	Regulator ground	
4	GNDA_SC	P	Oscillator ground	
5	GNDA_PLL	P	PLL ground	
6	VCC18V_PLL	P	PLL power in	
11,18,23, 30,36,43, 50,55,62, 70,77,83, 88,95	VCC_K	P	Core Power input	
17,22,29, 35,42,49, 61,69,76, 82,87,94 100	GN_D_K	P	Core Power ground	
27,41,57, 67,79,91	VCC_IO	P	I/O Power Input	
8,46,63, 74,89	GN_D_IO	P	I/O ground	

# FSA506 Application Note

---

## 4.1.2 Output Pins

Pin-No	Symbol	I/O	Description	Remark
59	X_VSYNC	O	Vertical sync signal output pin	
60	X_HSYNC	O	Horizontal sync signal output pin	
64	X_ODE	O	Data Enable Output Pin	
65	X_OCLK	O	Dot Clock Output Pin	
66	DO00	O	Display Data Output Pin	
68	DO01	O		
71	DO02	O		
72	DO03	O		
73	DO04	O		
75	DO05	O		
78	DO06	O		
80	DO07	O		
81	DO08	O		
84	DO09	O		
85	DO10	O		
86	DO11	O		
90	DO12	O		
92	DO13	O		
93	DO14	O		
96	DO15	O		
97	DO16	O		
98	DO17	O		
99	DCON	O	It is the GPIO pin for control the external DC/DC On off	

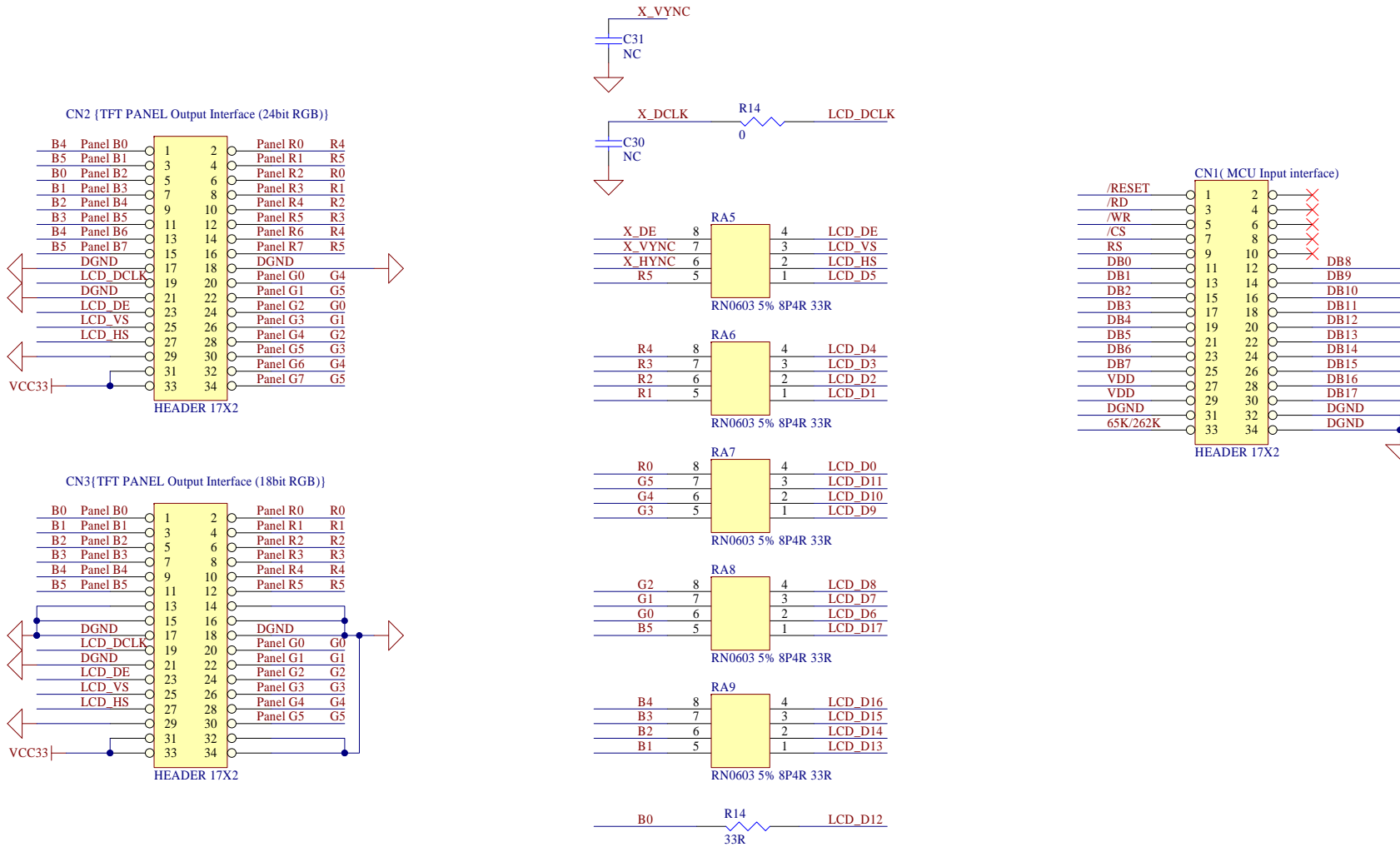
# FSA506 Application Note

## 4.1.3 Input Pins

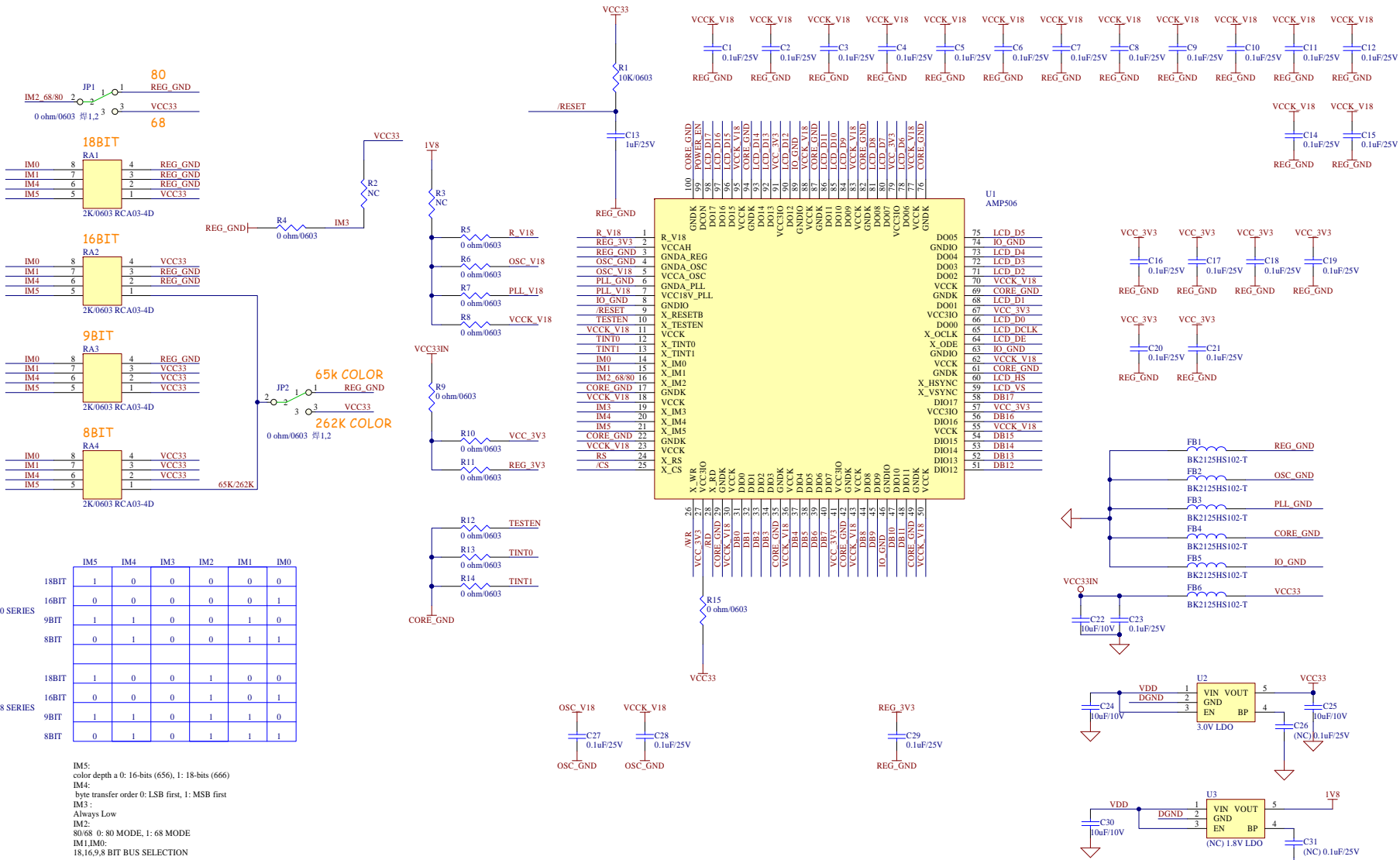
Pin-No	Symbol	I/O	Description	Remark																										
10	X_TESTEN	I	Test mode enable Pin. Connect to ground for normal operation																											
12	X_TIN0	I	Test mode reserved. Connect to ground for normal operation																											
13	X_TIN1	I	Test mode reserved. Connect to ground for normal operation																											
14	X_IM0	I	Interface select pins																											
			<table border="1"> <thead> <tr> <th colspan="2">X_IM[3:0]</th> <th colspan="2">MPU interface mode</th> </tr> </thead> <tbody> <tr> <td rowspan="4">0</td> <td rowspan="2">0</td> <td>0</td> <td>80-18bit bus interface</td> </tr> <tr> <td>1</td> <td>80-16bit bus interface</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>80-9bit bus interface</td> </tr> <tr> <td>1</td> <td>80-8bit bus interface</td> </tr> <tr> <td rowspan="4">1</td> <td rowspan="2">0</td> <td>0</td> <td>68-18bit bus interface</td> </tr> <tr> <td>1</td> <td>68-16bit bus interface</td> </tr> <tr> <td rowspan="2">1</td> <td>0</td> <td>68-9bit bus interface</td> </tr> <tr> <td>1</td> <td>68-8bit bus interface</td> </tr> </tbody> </table>	X_IM[3:0]		MPU interface mode		0	0	0	80-18bit bus interface	1	80-16bit bus interface	1	0	80-9bit bus interface	1	80-8bit bus interface	1	0	0	68-18bit bus interface	1	68-16bit bus interface	1	0	68-9bit bus interface	1	68-8bit bus interface	
X_IM[3:0]		MPU interface mode																												
0	0	0	80-18bit bus interface																											
		1	80-16bit bus interface																											
	1	0	80-9bit bus interface																											
		1	80-8bit bus interface																											
1	0	0	68-18bit bus interface																											
		1	68-16bit bus interface																											
	1	0	68-9bit bus interface																											
		1	68-8bit bus interface																											
15	X_IM1	I																												
16	X_IM2	I																												
19	X_IM3	I																												
20	X_IM4	I	Transport interface selection: X_IM4: Data byte transfer order <b>0: LSB first 1:MSB first</b>																											
21	X_IM5	I	X_IM5 : Data type <b>0:16-bits data (565) 1:18-bits data (666)</b>																											
24	X_RS	I	Register / Data select pin																											
25	X_CS	I	Chip select low active																											
26	X_WR	I	80mode : /WR low active 68mode : E signal latch on rising edge																											
28	X_RD	I	80mode : /RD low active 68mode : R/W signal Hi: read, Lo:Write																											
31	X_DIO0	I/O	MCU Data Bus input pins																											
32	X_DIO1	I/O																												
33	X_DIO2	I/O		<table border="1"> <thead> <tr> <th>X_IM[1:0]</th> <th>Bus_type</th> <th>Valid data bus</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>18-bits</td> <td>X_DIO17~0</td> </tr> <tr> <td>01</td> <td>16-bits</td> <td>X_DIO15~0</td> </tr> <tr> <td>10</td> <td>9 bits</td> <td>X_DIO8~0</td> </tr> <tr> <td>11</td> <td>8 bits</td> <td>X_DIO7~0</td> </tr> </tbody> </table>	X_IM[1:0]	Bus_type	Valid data bus	00	18-bits	X_DIO17~0	01	16-bits	X_DIO15~0	10	9 bits	X_DIO8~0	11	8 bits	X_DIO7~0											
X_IM[1:0]	Bus_type	Valid data bus																												
00	18-bits	X_DIO17~0																												
01	16-bits	X_DIO15~0																												
10	9 bits	X_DIO8~0																												
11	8 bits	X_DIO7~0																												
34	X_DIO3	I/O																												
37	X_DIO4	I/O																												
38	X_DIO5	I/O																												
39	X_DIO6	I/O																												
40	X_DIO7	I/O																												
44	X_DIO8	I/O																												
45	X_DIO9	I/O																												
47	X_DIO10	I/O																												
48	X_DIO11	I/O																												
51	X_DIO12	I/O																												
52	X_DIO13	I/O																												
53	X_DIO14	I/O																												
54	X_DIO15	I/O																												
56	X_DIO16	I/O																												
58	X_DIO17	I/O																												

# FSA506 Application Note

## 5 Application Circuit



# FSA506 Application Note



# FSA506 Application Note

## 6 Jumper Setting

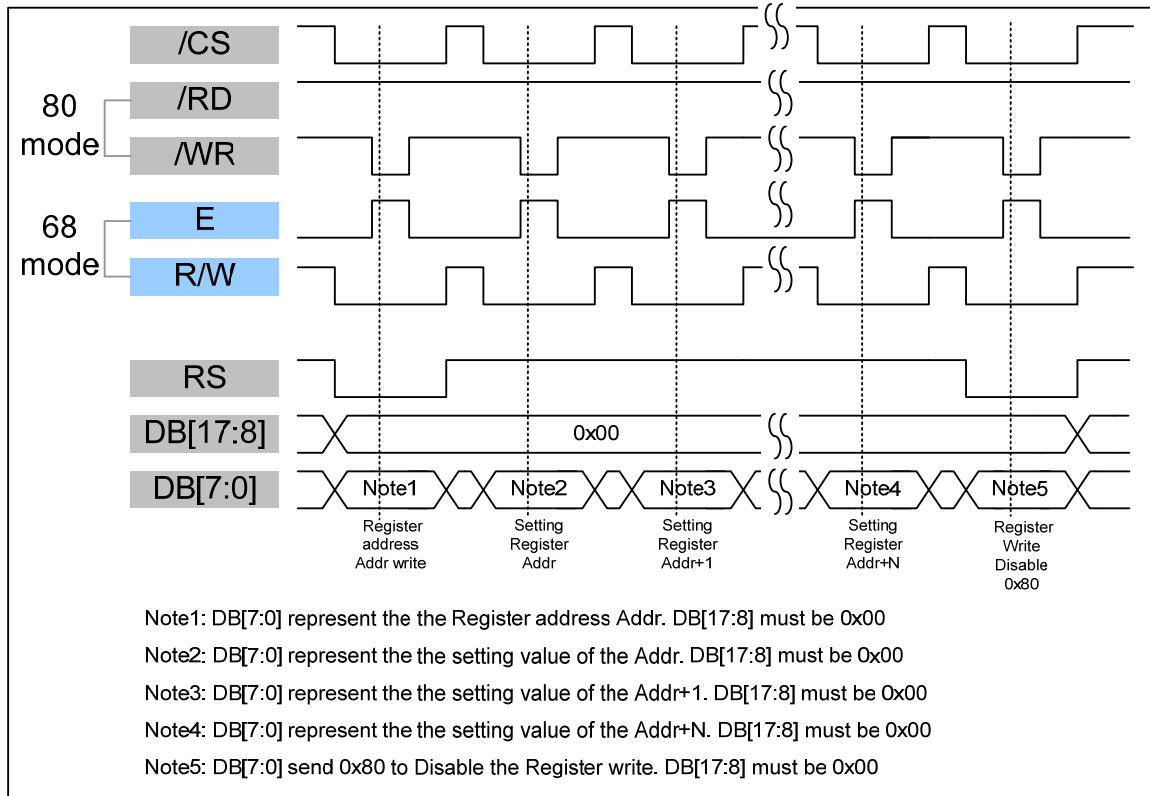
The user can select the MCU interface by change the Jumper & Resister Array.

Setting Interface Type	JP1	RA1	RA2	RA3	RA4	Remark
80-18Bit interface	1,2 short 2,3 open	2K ohm	OPEN	OPEN	OPEN	
80-16Bit interface	1,2 short 2,3 open	OPEN	2K ohm	OPEN	OPEN	
80-9Bit interface	1,2 short 2,3 open	OPEN	OPEN	2K ohm	OPEN	
<b>80-8Bit interface</b>	<b>1,2 short 2,3 open</b>	<b>OPEN</b>	<b>OPEN</b>	<b>OPEN</b>	<b>2K ohm</b>	<b>Default</b>
68-18Bit interface	1,2 open 2,3 short	2K ohm	OPEN	OPEN	OPEN	
68-16Bit interface	1,2 open 2,3 short	OPEN	2K ohm	OPEN	OPEN	
68-9Bit interface	1,2 open 2,3 short	OPEN	OPEN	2K ohm	OPEN	
68-8Bit interface	1,2 open 2,3 short	OPEN	OPEN	OPEN	2K ohm	

# FSA506 Application Note

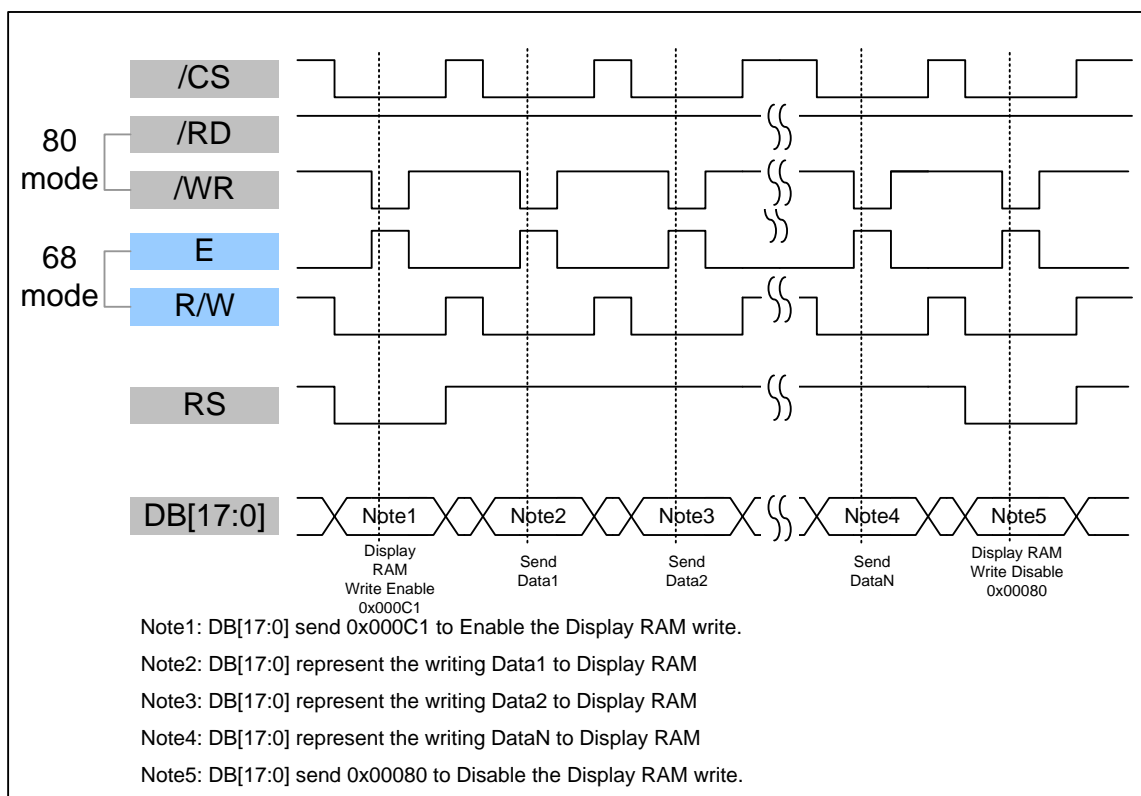
## 7 Interface Protocol

### 7.1 18Bit-80/68-Write to Command Register

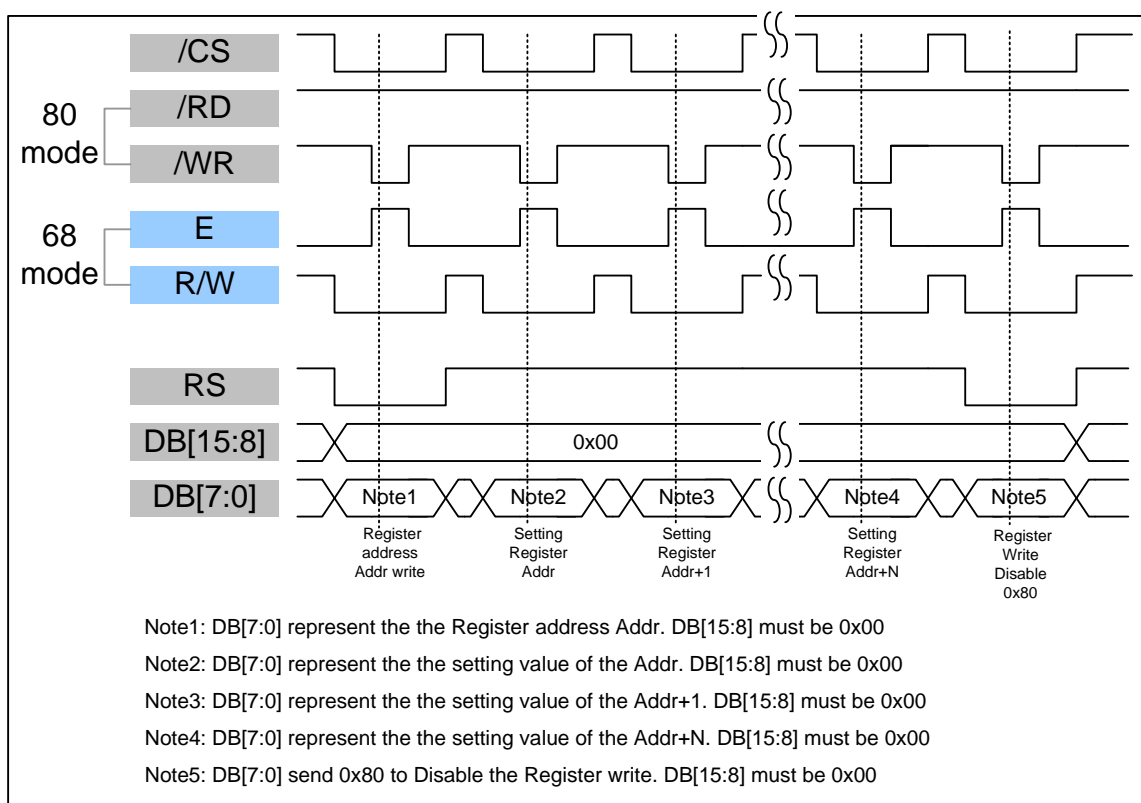


# FSA506 Application Note

## 7.2 18Bit-80/68-Write to Display RAM

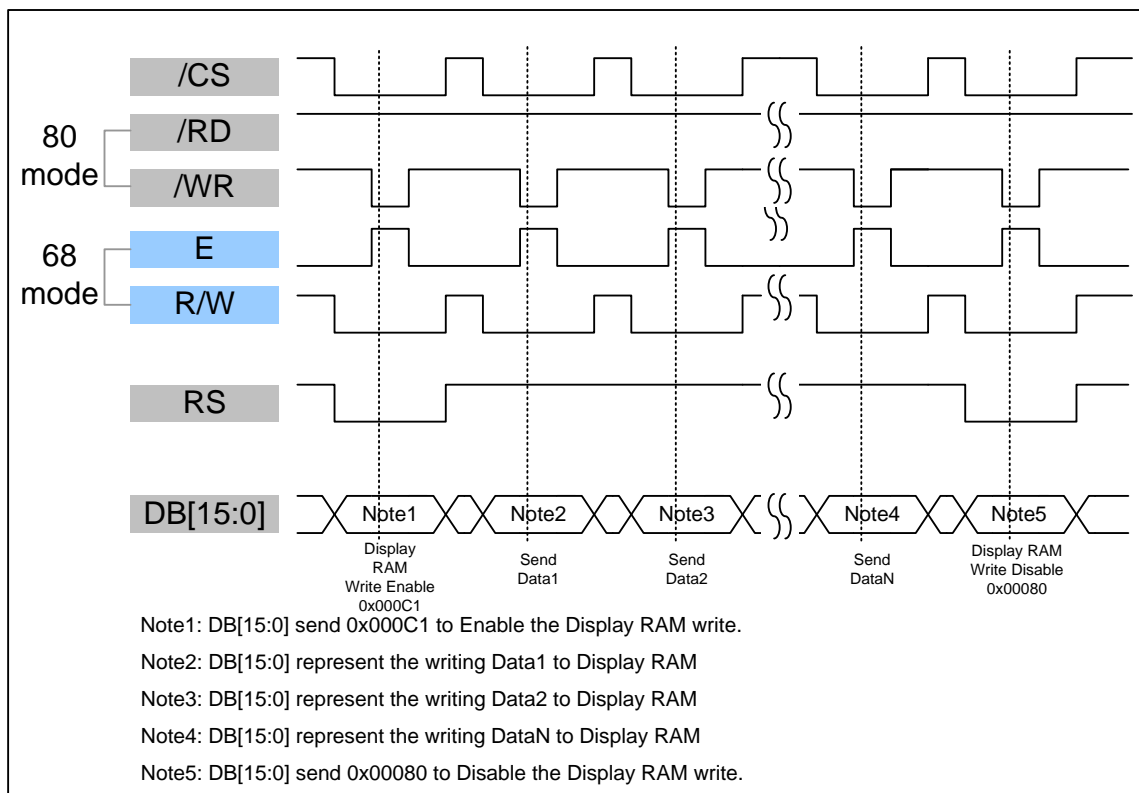


## 7.3 16Bit-80/68- Write to Command Register



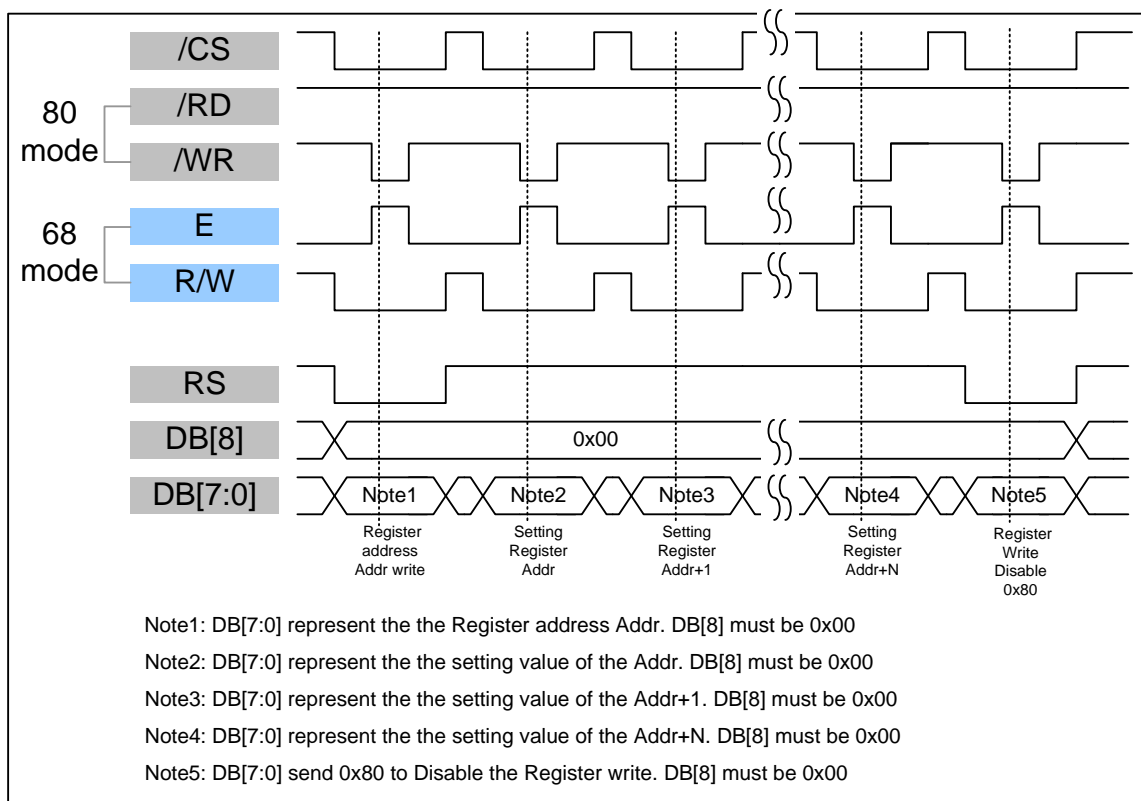
# FSA506 Application Note

## 7.4 16Bit-80/68-Write to Display RAM

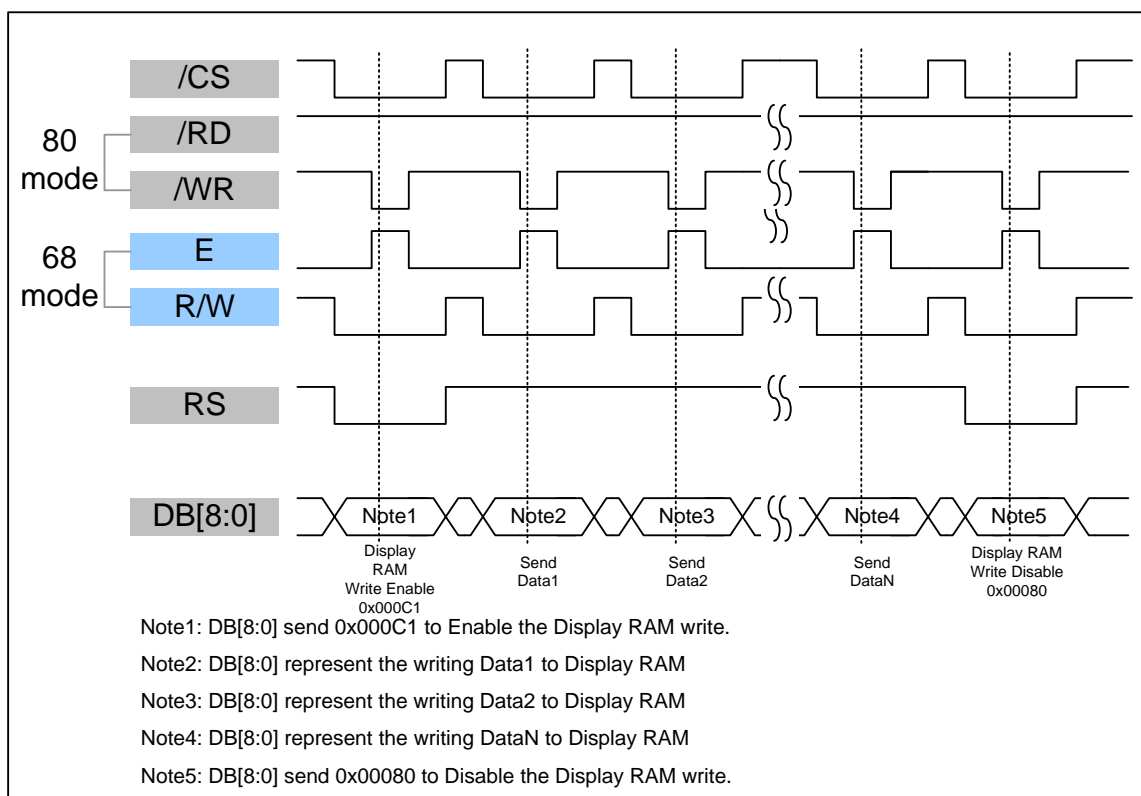


# FSA506 Application Note

## 7.5 9Bit-80/68- Write to Command Register

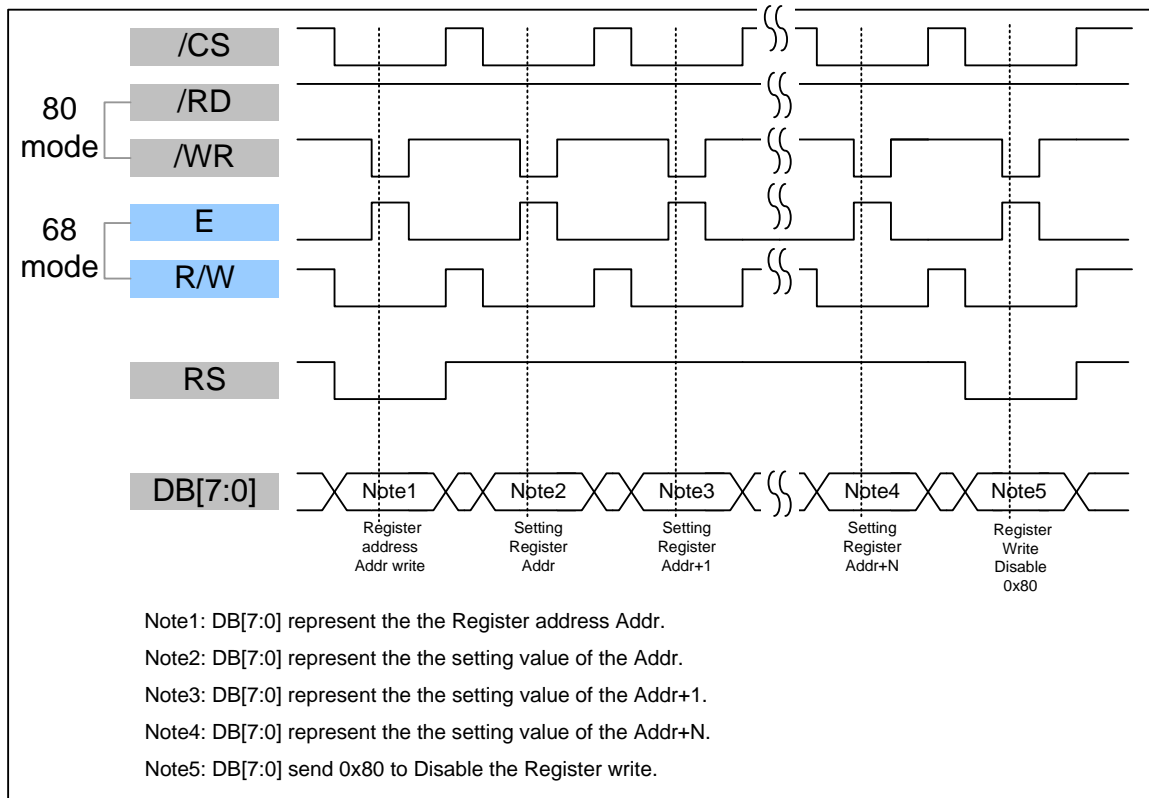


## 7.6 9Bit-80/68-Write to Display RAM



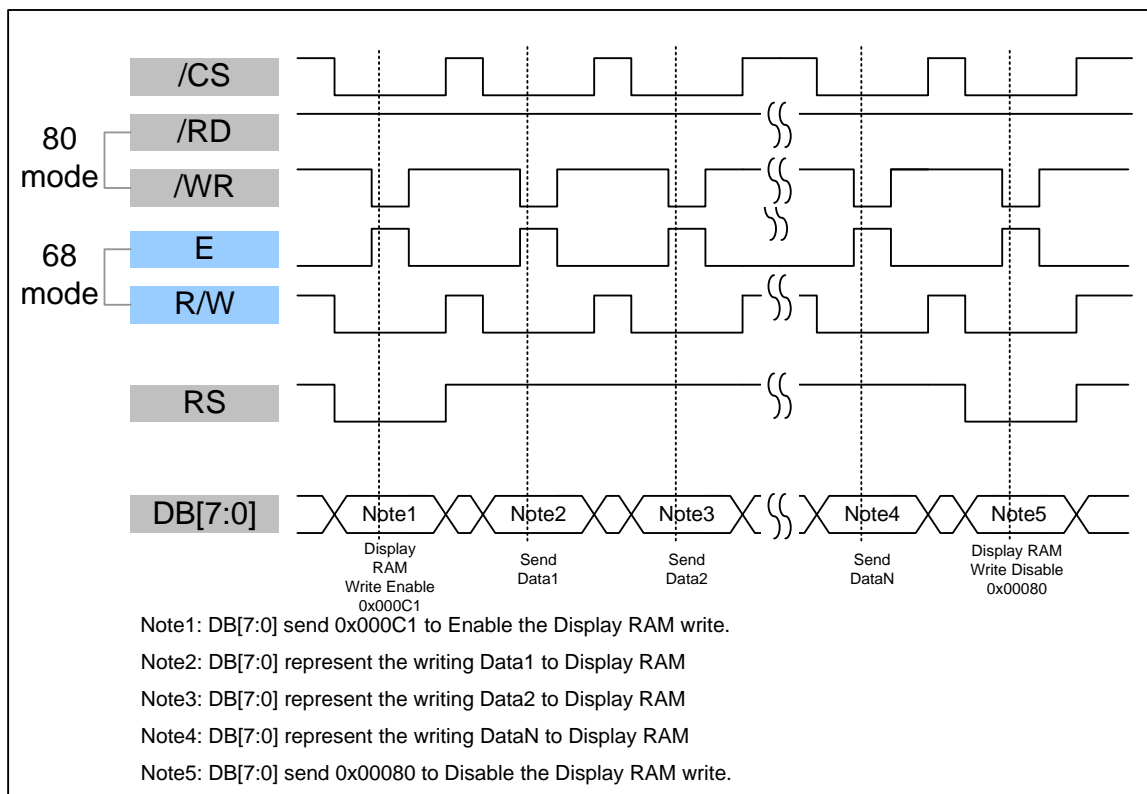
# FSA506 Application Note

## 7.7 8Bit-80/68- Write to Command Register



# FSA506 Application Note

## 7.8 8Bit-80/68-Write to Display RAM



# FSA506 Application Note

## 7.9 Data transfer order Setting

### 7.9.1 18 bit interface 262K color only (IM5: Pin 65K/262K =High)

DB	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R5	R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B5	B4	B3	B2	B1	B0

### 7.9.2 16 bit interface 65K color (IM5: Pin 65K/262K =Low)

DB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	R4	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B4	B3	B2	B1	B0

### 7.9.3 16 bit interface 262K color (IM5: Pin 65K/262K =High, IM4=Low)

DB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1 <sup>st</sup> data	R3	R2	R1	R0	G5	G4	G3	G2	G1	G0	B5	B4	B3	B2	B1	B0
2 <sup>nd</sup> data	X	X	X	X	X	X	X	X	X	X	X	X	X	X	R5	R4

### 7.9.4 9 bit interface 262K color only (IM5: Pin 65K/262K =High)

DB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1 <sup>st</sup> data	X	X	X	X	X	X	X	R5	R4	R3	R2	R1	R0	G5	G4	G3
2 <sup>nd</sup> data	X	X	X	X	X	X	X	G2	G1	G0	B5	B4	B3	B2	B1	B0

### 7.9.5 8 bit interface 65K color (IM5: Pin 65K/262K =Low)

DB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1 <sup>st</sup> data	X	X	X	X	X	X	X	X	R4	R3	R2	R1	R0	G5	G4	G3
2 <sup>nd</sup> data	X	X	X	X	X	X	X	X	G2	G1	G0	B4	B3	B2	B1	B0

### 7.9.6 8 bit interface 262K color (IM5: Pin 65K/262K =High)

DB	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1 <sup>st</sup> data	X	X	X	X	X	X	X	X							R5	R4
2 <sup>nd</sup> data	X	X	X	X	X	X	X	X	R3	R2	R1	R0	G5	G4	G3	G2
3 <sup>rd</sup> data	X	X	X	X	X	X	X	X	G1	G0	B5	B4	B3	B2	B1	B0

# FSA506 Application Note

## 8 Register Depiction

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
00	00	MSB of X-axis start position								
Description	set the horizontals start position of display active region									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
01	00	LSB of X-axis start position								
Description	set the horizontals start position of display active region									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
02	01	MSB of X-axis end position								
Description	set the horizontals end position of display active region									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
03	3F	LSB of X-axis end position								
Description	set the horizontals end position of display active region									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
04	00	MSB of Y-axis start position								
Description	set the vertical start position of display active region									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
05	00	LSB of Y-axis start position								
Description	Set the vertical start position of display active region									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
06	00	MSB of Y-axis end position								
Description	set the vertical end position of display active region									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
07	EF	LSB of Y-axis end position								
Description	Set the vertical end position of display active region									

To simplify the address control of display RAM access, the window area address function allows for writing data only within a window area of display RAM specified by

# FSA506 Application Note

---

registers REG[00]~REG[07].

After writing data to the display RAM, the Address counter will be increased within setting window address-range which is specified by

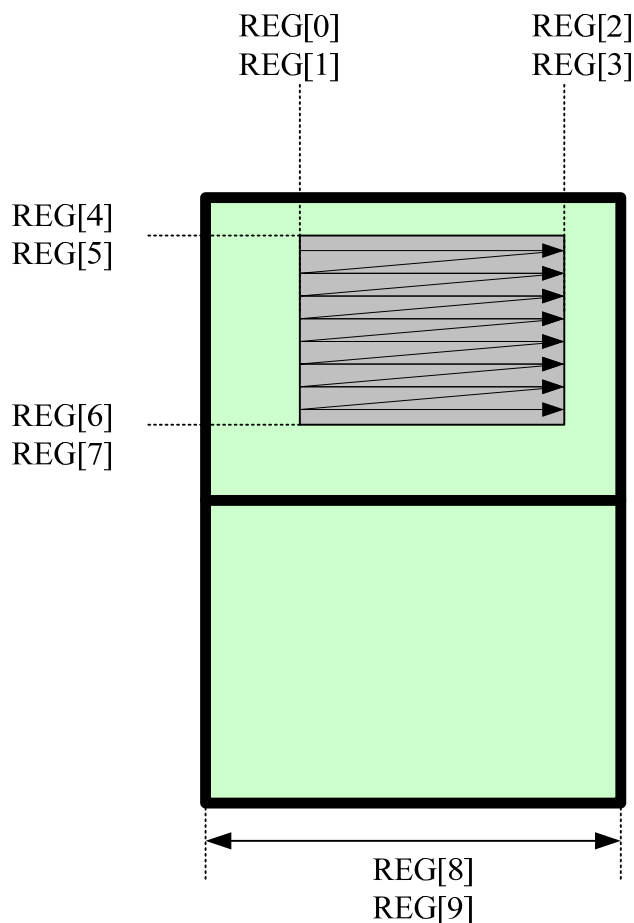
MIN X address (REG[0] & REG[1])

MAX X address (REG[2] & REG[3])

MIN Y address (REG[4] & REG[5])

MAX Y address (REG[6] & REG[7])

Therefore, data can be written consecutively without thinking the data address.



# FSA506 Application Note

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
08	01	X	X	X	X	X	X	_PanelXSize H_Byte[1:0]		
Description	Set the panel X size									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
09	40	_PanelXSize L_Byte[7:0]								
Description	Set the panel X size									

The register REG[08] and REG[09] is use to calculate the RAM address. If you want to use the TFT as Landscape mode (320x240), the REG[08] & RGE[09] must set to 320. If you want to use the TFT as Portrait mode (240x320), the REG[08] & RGE[09] must set to 240.

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0A	00	X	X	X	X	X	[17:16] bits of memory write start address			
Description	Memory write start address									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0B	00	[15:8] bits of memory write start address								
Description	Memory write start address									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0C	00	[7:0] bits of memory write start address								
Description	Memory write start address									

# FSA506 Application Note

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x10	0x0D	Bit_SWAP	OUT_TEST	BUS_SEL		Blanking	P/S_SEL	CLK_SEL		
Description	"0x10_Clk_sel[1:0]" : The TFT controller built-in 40Mhz PLL clock. These bits are for select the TFT panel dot clock frequency. 00 : 20Mhz 01: 10Mhz 02: 5 Mhz									
	"0x10_ps_sel[2]" : The TFT controller support parallel and serial RGB interface. These bits are for select the output timing. 0 : serial Panel 1: Parallel panel									
	"0x10_blanking_tmp[3]" 0 : OFF (blanking) 1: ON ( normal operation)									
	"0x10_bus_sel[5:4]" : It only for serial Panel 00=R , 01=G , 10=B									
	"0x10_out_test[6]" : Self test 0 : normal operation 1: for test (don't use for normal operation) When set the bit to "1" , the Rout=(Reg 2a[6:0]) Gout=(Reg 2b[6:0]) Bout=(Reg 2c[6:0])									
	"0x10_bit_swap[7]" : 0-normal The default setting is suitable for AM320240N1. Don't need to modify it.									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x11	00	X	X	EVEN			_ODD			
Description	" Even line of serial panel data out sequence or data bus order of parallel panel 000: <b>RGB</b> 001: RBG 010: GRB 011: GBR 100: BRG 101: BGR Others: reserved									
	Odd line of serial panel data out sequence 000: RGB 001: RBG 010: GRB 011: GBR 100: BRG 101: <b>BGR</b> Others: reserved <b>Must Set to 0x05</b>									

# FSA506 Application Note

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark	
0x12	00									Hsync_stH_Byte[3:0]	
Description	For TFT output timing adjust: Hsync start position H-Byte										
0x13	00									Hsync_stL_Byte[7:0]	
Description	For TFT output timing adjust: Hsync start position L-Byte										
0x14	00									Hsync_pwH_Byte[3:0]	
Description	For TFT output timing adjust: Hsync pulse width H-Byte										
0x15	10									Hsync_pwL_Byte[7:0]	
Description	For TFT output timing adjust: Hsync pulse width L-Byte										
0x16	00									Hact_stH_Byte[3:0]	
Description	For TFT output timing adjust: DE pulse start position H-Byte										
0x17	38									Hact_stL_Byte[7:0]	
Description	For TFT output timing adjust: DE pulse start position L-Byte										
0x18	01									Hact_pwH_Byte[3:0]	
Description	For TFT output timing adjust: DE pulse width H-Byte										
0x19	40									Hact_pwL_Byte[7:0]	
Description	For TFT output timing adjust: DE pulse width L-Byte										

# FSA506 Application Note

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x1A	01					HtotalH_Byte[3:0]				
Description	For TFT output timing adjust: Hsync total clocks H-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x1B	B8	HtotalL_Byte[7:0]								
Description	For TFT output timing adjust: Hsync total clocks H-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x1C	00					Vsync_stH_Byte[3:0]				
Description	For TFT output timing adjust: Vsync start position H-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x1D	00	Vsync_stL_Byte[7:0]								
Description	For TFT output timing adjust: Vsync start position L-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x1E	00					Vsync_pwH_Byte[3:0]				
Description	For TFT output timing adjust: Vsync pulse width H-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x1F	08	Vsync_pwL_Byte[7:0]								
Description	For TFT output timing adjust: Vsync pulse width L-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x20	00					Vact_stH_Byte[3:0]				
Description	For TFT output timing adjust: Vertical DE pulse start position H-Byte									

# FSA506 Application Note

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x21	12	Vact_stL_Byte[7:0]								
Description	For TFT output timing adjust: Vertical DE pulse start position L-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x22	00	Vact_pwH_Byte[3:0]								
Description	For TFT output timing adjust: Vertical Active width H-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x23	F0	Vact_pwL_Byte[7:0]								
Description	For TFT output timing adjust: Vertical Active width H-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x24	01	VtotalH_Byte[3:0]								
Description	For TFT output timing adjust: Vertical total width H-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x25	09	VtotalL_Byte[7:0]								
Description	For TFT output timing adjust: Vertical total width L-Byte									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
26	00	X	X	X	X	X	[17:16] bits of memory read start address			
Description	Memory read start address									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
27	00	[15:8] bits of memory write start address								
Description	Memory read start address									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
28	00	[7:0] bits of memory write start address								
Description	Memory read start address									

# FSA506 Application Note

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark	
29	00	[7:1] Reversed									
Description	[0] Load output timing related setting (H sync., V sync. and DE) to take effect										
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark	
0x2A	00	X	TestPatternRout[6:0]								
Description	When " REG[0x10]_out_test[6]" : Self test =1 ; The Rout data equal to TestPatternRout[6:0]										
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark	
0x2B	00	X	TestPatternGout[6:0]								
Description	When " REG[0x10]_out_test[6]" : Self test =1 ; The Gout data equal to TestPatternGout[6:0]										
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark	
0x2C	00	X	TestPatternBout[6:0]								
Description	When " REG[0x10]_out_test[6]" : Self test =1 ; The Bout data equal to TestPatternBout[6:0]										

If you set the " REG[0x10]\_out\_test[6]" : Self test =1 , the TFT controller will skip the connect of the display RAM. The Output port will send the REG[2A] ,REG[2B],REG[2C] data.

REG[2A]=0x3F  
REG[2B]=0x00  
REG[2C]=0x00

REG[2A]=0x00  
REG[2B]=0x3F  
REG[2C]=0x00

REG[2A]=0x00  
REG[2B]=0x00  
REG[2C]=0x3F

# FSA506 Application Note

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
0x2D	00	X	X	X	X	[3]	Rising/falling edge[2]	_rotate [1:0]		
Description	[3] Output pin X_DCON level control ; TFT Power ON/OFF control 0: TFT POWER circuit OFF 1: TFT POWER circuit ON									
	Rising/falling edge[2] : 0: The RGB out put data are on the Rising edge of the DCLK. 1: The RGB out put data are on the Falling edge of the DCLK.									
	_rotate [1:0]: 00 : rotate 0 degree 01 : rotate 90 degree 10 : rotate 270 degree 11 : rotate 180 degree									

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
30	00	X	X	X	X	X	_H byte H-Offset[3:0]			
Description	Set the Horizontal offset									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
31	00	_L byte H-Offset[7:0]								
Description	Set the Horizontal offset									

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
32	00	X	X	X	X	X	_H byte V-Offset[3:0]			
Description	Set the Vertical offset									
Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
33	00	_L byte V-Offset[7:0]								
Description	Set the Vertical offset									

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark	
34	00	[7:4] Reserved					_H byte H-def[3:0]				
Description	[3:0] MSB of image horizontal physical resolution in memory										

# FSA506 Application Note

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
35	40	_L byte H-def[7:0]								
Description	[7:0] LSB of image horizontal physical resolution in memory									

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
36	01	[7:4] Reserved					_H byte V-def[3:0]			
Description	[3:0] MSB of image vertical physical resolution in memory									

Register Address (Hex)	Default (Hex)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Remark
37	E0	_L byte V-def[7:0]								
Description	[7:0] LSB of image vertical physical resolution in memory									

The total RAM size is 640x240x18bit. The user can arrange the Horizontal ram size by REG[34],REG[35] and the Vertical ram size by REG[36],REG[37].

EX: 320x480x18bit REG[34]=0x01 , REG[35]=0x40 , REG[36]=0x01 , REG[37]=0xE0

EX: 640x240x18bit. REG[34]=0x02 , REG[35]=0x80 , REG[36]=0x00 , REG[37]=0xF0

## 9 Reference Initial code :

9.1 8bit-80 interface mode , 65K color , Panel:320xRGBx240 , LandScape

```
void main(void)
{
    Initial_FSA506 ( );
    Full_386SCR(0xf800);
    Full_386SCR(0x07e0);
    Full_386SCR(0x001f);
}
```

# FSA506 Application Note

---

```
void FSA506_80Mode_Command_SendAddress(BYTE Addr)
```

```
{  
    SET_nRD;           // /RD=1  
    CLR_RS;           // RS=0  
    CLR_CS1;          // /CS=0  
    CLR_nWRL;         // /WR=0  
    DB8OUT(Addr);     // Data Bus OUT  
    SET_nWRL;         // /WR=1    /  
    SET_RS;           // RS=1  
    SET_CS1;          // CS=1  
}
```

```
void FSA506_80Mode_Command_SendData(BYTE Data)
```

```
{  
    SET_nRD;  
    SET_RS;  
    CLR_CS1;  
    CLR_nWRL;  
    DB8OUT(Data);  
    SET_nWRL;  
    SET_RS;  
    SET_CS1;  
}
```

```
void FSA506_Command_Write(uint8 CMD_Address,uint8 CMD_Value)
```

```
{  
    FSA506_80Mode_Command_SendAddress(CMD_Address);  
    FSA506_80Mode_Command_SendData(CMD_Value);  
}
```

```
void FSA506_80Mode_16Bit_Memory_SendData(uint16 Dat16bit)
```

```
{  
    SET_nRD;  
    SET_RS;  
    CLR_CS1;
```

# FSA506 Application Note

---

```
CLR_nWRL;
DB8OUT(Dat16bit>>8);
SET_nWRL; // Low to High Latch Data to FSA506 Buffer
SET_CS1;

SET_nRD;
SET_RS;
CLR_CS1;
CLR_nWRL;
DB8OUT(Dat16bit);
SET_nWRL; // Low to High Latch Data to FSA506 Buffer
SET_CS1;

}

void Initial_FSA506(void)
{
    FSA506_Command_Write(0x40,0x12); /*[7:6] Reserved
                                     [5] PLL control pins to select out frequency range
                                     0: 20MHz ~ 100MHz 1: 100MHz ~ 300MHz
                                     [4] Reserved [3] Reserved
                                     [2:1] Output Driving Capability
                                     00: 4mA 01: 8mA 10: 12mA 11: 16mA
                                     [0] Output slew rate
                                     0: Fast 1: Slow
                                     */
    FSA506_Command_Write(0x41,0x01); //Set PLL=40Mhz * (0x42) / (0x41)
    FSA506_Command_Write(0x42,0x01); //0x41 [7:6] Reserved [5:0] PLL Programmable
                                     pre-divider, 6bit(1~63)
                                     //0x42 [7:6] Reserved [5:0] PLL Programmable loop
                                     divider, 6bit(1~63)

    FSA506_Command_Write(0x00,0x00); // MSB of horizontal start coordinate value
    FSA506_Command_Write(0x01,0x00); // LSB of horizontal start coordinate value
    FSA506_Command_Write(0x02,0x01); // MSB of horizontal end coordinate value
    FSA506_Command_Write(0x03,0x3F); // LSB of horizontal end coordinate value
    FSA506_Command_Write(0x04,0x00); // MSB of vertical start coordinate value
    FSA506_Command_Write(0x05,0x00); // LSB of vertical start coordinate value
    FSA506_Command_Write(0x06,0x01); // MSB of vertical end coordinate value
```

# FSA506 Application Note

---

FSA506\_Command\_Write(0x07,0x3F); // LSB of vertical end coordinate value

FSA506\_Command\_Write(0x08,0x01); // MSB of input image horizontal resolution  
FSA506\_Command\_Write(0x09,0x40); // LSB of input image horizontal resolution  
FSA506\_Command\_Write(0x0a,0x00); //[17:16] bits of memory write start address  
FSA506\_Command\_Write(0x0b,0x00); //[15:8] bits of memory write start address  
FSA506\_Command\_Write(0x0c,0x00); //[7:0] bits of memory write start address

FSA506\_Command\_Write(0x10,0x0D); /\*[7] Output data bits swap 0: Normal 1:Swap  
[6] Output test mode enable 0: disable 1: enable  
[5:4] Serial mode data out bus selection  
00: X\_ODATA17 ~ X\_ODATA12 active ,others are set to 0  
01: X\_ODATA11 ~ X\_ODATA06 active ,others are set to 0  
10: X\_ODATA05 ~ X\_ODATA00 active ,others are set to 0  
11: reserved  
[3] Output data blanking  
0: set output data to 0 **1: Normal display**  
[2] Parallel or serial mode selection  
0: serial data out **1: parallel data output**  
[1:0] Output clock selection  
00: system clock divided by 2  
**01: system clock divided by 4**  
10: system clock divided by 8  
11: reserved \*/

FSA506\_Command\_Write(0x11,0x05);  
/\*[7] Reserved  
[6:4] Even line of serial panel data out sequence or data bus order of parallel panel  
000: RGB 001: RBG 010: GRB 011: GBR 100: BRG 101: BGR  
Others: reserved  
[3] Reversed  
[2:0] Odd line of serial panel data out sequence  
000: RGB 001: RBG 010: GRB 011: GBR 100: BRG **101: BGR**  
Others: reserved \*/

FSA506\_Command\_Write(0x12,0x00); // [3:0] MSB of output H sync. pulse start position  
FSA506\_Command\_Write(0x13,0x00); //[7:0] LSB of output H sync. pulse start position  
FSA506\_Command\_Write(0x14,0x00); // [3:0] MSB of output H sync. pulse width  
FSA506\_Command\_Write(0x15,0x10); // [7:0] LSB of output H sync. pulse width  
FSA506\_Command\_Write(0x16,0x00); //[3:0] MSB of output DE horizontal start position

# FSA506 Application Note

---

FSA506\_Command\_Write(0x17,0x38); // [7:0] LSB of output DE horizontal start position

FSA506\_Command\_Write(0x18,0x01); // [3:0] MSB of output DE horizontal active region in pixel

FSA506\_Command\_Write(0x19,0x40); // [7:0] LSB of output DE horizontal active region in pixel

FSA506\_Command\_Write(0x1a,0x01); // [7:4] Reserved [3:0] MSB of output H total in pixel

FSA506\_Command\_Write(0x1b,0xb8); // [7:0] LSB of output H total in pixel

FSA506\_Command\_Write(0x1c,0x00); // [3:0] MSB of output V sync. pulse start position

FSA506\_Command\_Write(0x1d,0x00); // [7:0] of output V sync. pulse start position

FSA506\_Command\_Write(0x1e,0x00); // [7:4] Reserved [3:0] MSB of output V sync. pulse width

FSA506\_Command\_Write(0x1f,0x08); // [7:0] LSB of output V sync. pulse width

FSA506\_Command\_Write(0x20,0x00); // [3:0] MSB of output DE vertical start position

FSA506\_Command\_Write(0x21,0x12); // [7:0] LSB of output DE vertical start position

FSA506\_Command\_Write(0x22,0x00); // [3:0] MSB of output DE vertical active region in line

FSA506\_Command\_Write(0x23,0xf0); // [7:0] LSB of output DE vertical active region in line

FSA506\_Command\_Write(0x24,0x01); // [7:4] Reversed [3:0] MSB of output V total in line

FSA506\_Command\_Write(0x25,0x09); // [7:0] LSB of output V total in line

FSA506\_Command\_Write(0x26,0x00); // [17:16] bits of memory read start address

FSA506\_Command\_Write(0x27,0x00); // [7:0] [15:8] bits of memory read start address

FSA506\_Command\_Write(0x28,0x00); // [7:0] [7:0] bits of memory read start address

FSA506\_Command\_Write(0x29,0x01);  
// [7:1] Reversed [0] Load output timing related setting (H sync., V sync. and DE) to take effect

FSA506\_Command\_Write(0x2d,0x08); /\* [7:4] Reserved  
[3] Output pin X\_DCON level control  
[2] Output clock inversion 0: Normal 1: Inverse  
[1:0] Image rotate  
00: 0° 01: 90° 10: 270° 11: 180°  
\*/

FSA506\_Command\_Write(0x30,0x00); // [7:4] Reserved [3:0] MSB of image horizontal shift value

FSA506\_Command\_Write(0x31,0x00); // [7:0] LSB of image horizontal shift value

FSA506\_Command\_Write(0x32,0x00); // [7:4] Reserved [3:0] MSB of image vertical shift value

FSA506\_Command\_Write(0x33,0x00); // [7:0] LSB of image vertical shift value

FSA506\_Command\_Write(0x34,0x01);  
// [3:0] MSB of image horizontal physical Resolution in memory

FSA506\_Command\_Write(0x35,0x40);  
// [7:0] LSB of image horizontal physical resolution in memory

FSA506\_Command\_Write(0x36,0x01);  
// [7:4] Reserved [3:0] MSB of image vertical physical resolution in memory

FSA506\_Command\_Write(0x37,0xe0);

# FSA506 Application Note

---

```
//[7:0] LSB of image vertical physical resolution in memory
```

```
}
```

```
void FSA506_WindowSet(uint16 S_X,uint16 S_Y,uint16 E_X,uint16 E_Y)
```

```
{
```

```
    FSA506_80Mode_Command_SendAddress(0x00);
    FSA506_80Mode_Command_SendData((S_X)>>8);
    FSA506_80Mode_Command_SendData(S_X);
    FSA506_80Mode_Command_SendData((E_X-1)>>8);
    FSA506_80Mode_Command_SendData(E_X-1);
    FSA506_80Mode_Command_SendData(S_Y>>8);
    FSA506_80Mode_Command_SendData(S_Y);
    FSA506_80Mode_Command_SendData((E_Y-1)>>8);
    FSA506_80Mode_Command_SendData(E_Y-1);
```

```
}
```

```
void Full_386SCR(uint16 Dat16bit)
```

```
{
```

```
    int32 k,l;
```

```
    FSA506_WindowSet(0,0,Resolution_X,Resolution_Y);
```

```
    FSA506_80Mode_Command_SendAddress(0xc1); // _DisplayRAM_WriteEnable_
```

```
    for(k=0;k<240*2;k++)
```

```
    {
```

```
        for(l=0;l<320;l++)
```

```
        {
```

```
            FSA506_80Mode_16Bit_Memory_SendData(Dat16bit);
```

```
        }
```

```
    }
```

```
    FSA506_80Mode_Command_SendAddress(0x80); // _DisplayRAM_WriteDisable_
```

```
}
```

# FSA506 Application Note

---

9.2 8bit-80 interface mode , 65K color , Panel:320xRGBx240 , Portrait

```
void Initial_FSA506_90(void)
{
FSA506_Command_Write(0x40,0x12); /*[7:6] Reserved
                                [5] PLL control pins to select out frequency range
                                0: 20MHz ~ 100MHz 1: 100MHz ~ 300MHz
                                [4] Reserved [3] Reserved
                                [2:1] Output Driving Capability
                                00: 4mA 01: 8mA 10: 12mA 11: 16mA
                                [0] Output slew rate
                                0: Fast 1: Slow
                                */
FSA506_Command_Write(0x41,0x01); //Set PLL=40Mhz * (0x42) / (0x41)
FSA506_Command_Write(0x42,0x01);
//0x41 [7:6] Reserved [5:0] PLL Programmable pre-divider, 6bit(1~63)
//0x42 [7:6] Reserved [5:0] PLL Programmable loop divider, 6bit(1~63)

FSA506_Command_Write(0x00,0x00); // MSB of horizontal start coordinate value
FSA506_Command_Write(0x01,0x00); // LSB of horizontal start coordinate value
FSA506_Command_Write(0x02,0x01); // MSB of horizontal end coordinate value
FSA506_Command_Write(0x03,0x3F); // LSB of horizontal end coordinate value
FSA506_Command_Write(0x04,0x00); // MSB of vertical start coordinate value
FSA506_Command_Write(0x05,0x00); // LSB of vertical start coordinate value
FSA506_Command_Write(0x06,0x01); // MSB of vertical end coordinate value
FSA506_Command_Write(0x07,0x3F); // LSB of vertical end coordinate value
FSA506_Command_Write(0x08,0x00); // MSB of input image horizontal resolution
FSA506_Command_Write(0x09,0xf0);
// LSB of input image horizontal resolution Portrait X=240
FSA506_Command_Write(0x0a,0x00); //[17:16] bits of memory write start address
FSA506_Command_Write(0x0b,0x00); //[15:8] bits of memory write start address
FSA506_Command_Write(0x0c,0x00); //[7:0] bits of memory write start address

FSA506_Command_Write(0x10,0x0D);
```

# FSA506 Application Note

---

/\*[7] Output data bits swap 0: Normal 1:Swap

[6] Output test mode enable 0: disable 1: enable

[5:4] Serial mode data out bus selection

00: X\_ODATA17 ~ X\_ODATA12 active , others are set to zero

01: X\_ODATA11 ~ X\_ODATA06 active , others are set to zero

10: X\_ODATA05 ~ X\_ODATA00 active , others are set to zero

11: reserved

[3] Output data blanking

0: set output data to 0

1: Normal display

[2] Parallel or serial mode selection

0: serial data out 1: parallel data output[7:0] bits of memory write start address

[1:0] Output clock selection

00: system clock divided by 2

01: system clock divided by 4

10: system clock divided by 8

11: reserved \*/

FSA506\_Command\_Write(0x11,0x05);

/\*[7] Reserved

[6:4] Even line of serial panel data out sequence or data bus order of parallel panel

000: RGB

001: RBG

010: GRB

011: GBR

100: BRG

101: BGR

Others: reserved

[3] Reversed

[2:0] Odd line of serial panel data out sequence

000: RGB

001: RBG

010: GRB

011: GBR

100: BRG

101: BGR

Others: reserved

\*/

FSA506\_Command\_Write(0x12,0x00);

//[7:4] Reserved [3:0] MSB of output H sync. pulse start position

FSA506\_Command\_Write(0x13,0x00); //[7:0] LSB of output H sync. pulse start position

# FSA506 Application Note

---

FSA506\_Command\_Write(0x14,0x00); // [7:4] Reserved [3:0] MSB of output H sync. pulse width  
FSA506\_Command\_Write(0x15,0x10); // [7:0] LSB of output H sync. pulse width  
FSA506\_Command\_Write(0x16,0x00);  
// [7:4] Reserved [3:0] MSB of output DE horizontal start position  
FSA506\_Command\_Write(0x17,0x38); // [7:0] LSB of output DE horizontal start position  
FSA506\_Command\_Write(0x18,0x01);  
// [7:4] Reserved [3:0] MSB of output DE horizontal active region in pixel  
FSA506\_Command\_Write(0x19,0x40);  
// [7:0] LSB of output DE horizontal active region in pixel  
FSA506\_Command\_Write(0x1a,0x01); // [7:4] Reserved [3:0] MSB of output H total in pixel  
FSA506\_Command\_Write(0x1b,0xb8); // [7:0] LSB of output H total in pixel  
FSA506\_Command\_Write(0x1c,0x00);  
// [7:4] Reserved [3:0] MSB of output V sync. pulse start position  
FSA506\_Command\_Write(0x1d,0x00); // [7:0] of output V sync. pulse start position  
FSA506\_Command\_Write(0x1e,0x00); // [7:4] Reserved [3:0] MSB of output V sync. pulse  
width  
FSA506\_Command\_Write(0x1f,0x08); // [7:0] LSB of output V sync. pulse width  
FSA506\_Command\_Write(0x20,0x00);  
// [7:4] Reserved [3:0] MSB of output DE vertical start position  
FSA506\_Command\_Write(0x21,0x12); // [7:0] LSB of output DE vertical start position  
FSA506\_Command\_Write(0x22,0x00); // [7:4] Reserved [3:0] MSB of output DE vertical  
active region in line  
FSA506\_Command\_Write(0x23,0xf0); // [7:0] LSB of output DE vertical active region in line  
FSA506\_Command\_Write(0x24,0x01); // [7:4] Reversed [3:0] MSB of output V total in line  
FSA506\_Command\_Write(0x25,0x09); // [7:0] LSB of output V total in line  
FSA506\_Command\_Write(0x26,0x00);  
// [7:2] Reserved [1:0] [17:16] bits of memory read start address  
FSA506\_Command\_Write(0x27,0x00); // [7:0] [15:8] bits of memory read start address  
FSA506\_Command\_Write(0x28,0x00); // [7:0] [7:0] bits of memory read start address  
FSA506\_Command\_Write(0x29,0x01);  
// [7:1] Reversed [0] Load output timing related setting (H sync., V sync. and DE) to take effect

FSA506\_Command\_Write(0x2d,0x09);  
/\* [7:4] Reserved  
[3] Output pin X\_DCON level control

# FSA506 Application Note

---

[2] Output clock inversion      0: Normal 1: Inverse

[1:0] Image rotate

00: 0°   01: 90°   10: 270° 11: 180°

\*/

```
FSA506_Command_Write(0x30,0x00); //[7:4] Reserved [3:0] MSB of image horizontal shift value
```

```
FSA506_Command_Write(0x31,0x00); //[7:0] LSB of image horizontal shift value
```

```
FSA506_Command_Write(0x32,0x00); //[7:4] Reserved [3:0] MSB of image vertical shift value
```

```
FSA506_Command_Write(0x33,0x00); //[7:0] LSB of image vertical shift value
```

```
FSA506_Command_Write(0x34,0x00);
```

```
    //[7:4] Reserved [3:0] MSB of image horizontal physical resolution in memory
```

```
FSA506_Command_Write(0x35,0xf0);
```

```
    //[7:0] LSB of image horizontal physical resolution in memory
```

```
FSA506_Command_Write(0x36,0x02);
```

```
    //[7:4] Reserved [3:0] MSB of image vertical physical resolution in memory
```

```
FSA506_Command_Write(0x37,0x80);
```

```
    //[7:0] LSB of image vertical physical resolution in memory
```

```
}
```

# FSA506 Application Note

The TFT LCD controller default value is for 320xRGBx240 already. So we can start to write our data in a few steps:

Target: To write a 640x240 data to Display RAM and scroll the display data by change the Horizontal offset register.

9.3 Step 1: Make sure the interface Protocol.

9.4 Step 2: Define the Horizontal ram seize = 640 and Vertical ram size =240  
640x240x18bit. REG[34]=0x02 , REG[35]=0x80 , REG[36]=0x00 ,

REG[37]=0xF0

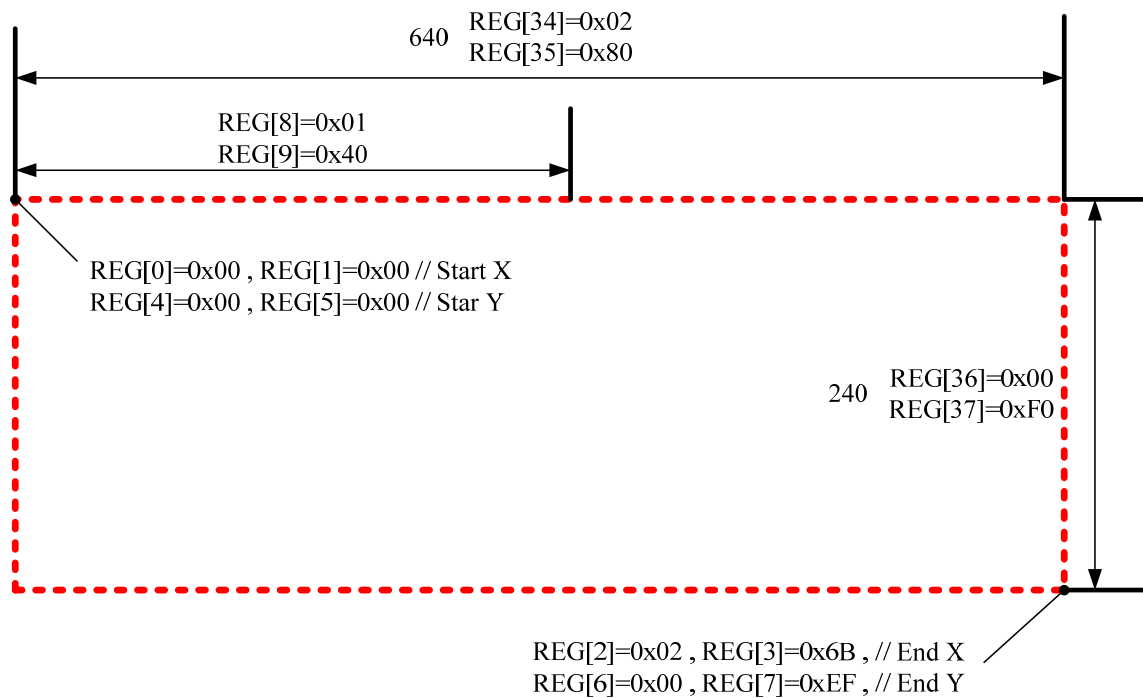
9.5 Step 3: Define the Panel X Size = 320

REG[8]=0x01 , REG[9]=0x40

9.6 Step4: Define the Write window. Start=(0,0) End=(619,239)

REG[0]=0x00 , REG[1]=0x00 , REG[2]=0x02 , REG[3]=0x6B , // Start X , End X

REG[4]=0x00 , REG[5]=0x00 , REG[6]=0x00 , REG[7]=0xEF , // Star Y ,End Y



# FSA506 Application Note

9.7 Step5: Write the 640x240x18 bit data consecutively



9.8 Step6: The display will show the following image.



9.9 Step7: Change the Horizontal offset to switch or scroll the display data. Set the Horizontal offset = 160 , REG[30]=00 REG[31]=A0 . You will see



# FSA506 Application Note

---

9.10 Step8: Change the Horizontal offset to switch or scroll the display data.

Set the Horizontal offset = 320 , REG[30]=01 REG[31]=40 . You will see

