Focus Display Solutions
www.focuslcd.com
162 W Boxelder Place, Suite 1
Chandler, AZ 85225
Telephone: (480)503-4295
Fax: (866)752-0165
Email: jont@focuslcd.com

Field Sequential Color (FSC) LCD

Contents

Specification Revision History

Doc. Version	Revision Description	Date
V1.0	Initial version	2008-10-21
V1.1	Modify pixel addr.	2008-10-22

1 General Description:

FS-LCD is a new type color display LCD.These patented displays use RGB backlights to display hues instead of color filters. As a result, our modules feature crisp coloration similar to TFT LCDS. Customer's MCU sending different Color Command to each pixel, which can display 8 colors (Red, Green, Blue, Yellow, Cyan, Purple, Black and Write)

FS-LCD can be applied to the Instrument, Meter, Home Appliance, Communication, Clock, Car Audio, HI-FI, Game Machine and Large Displays in Airport, Subway information display.

2 Features:

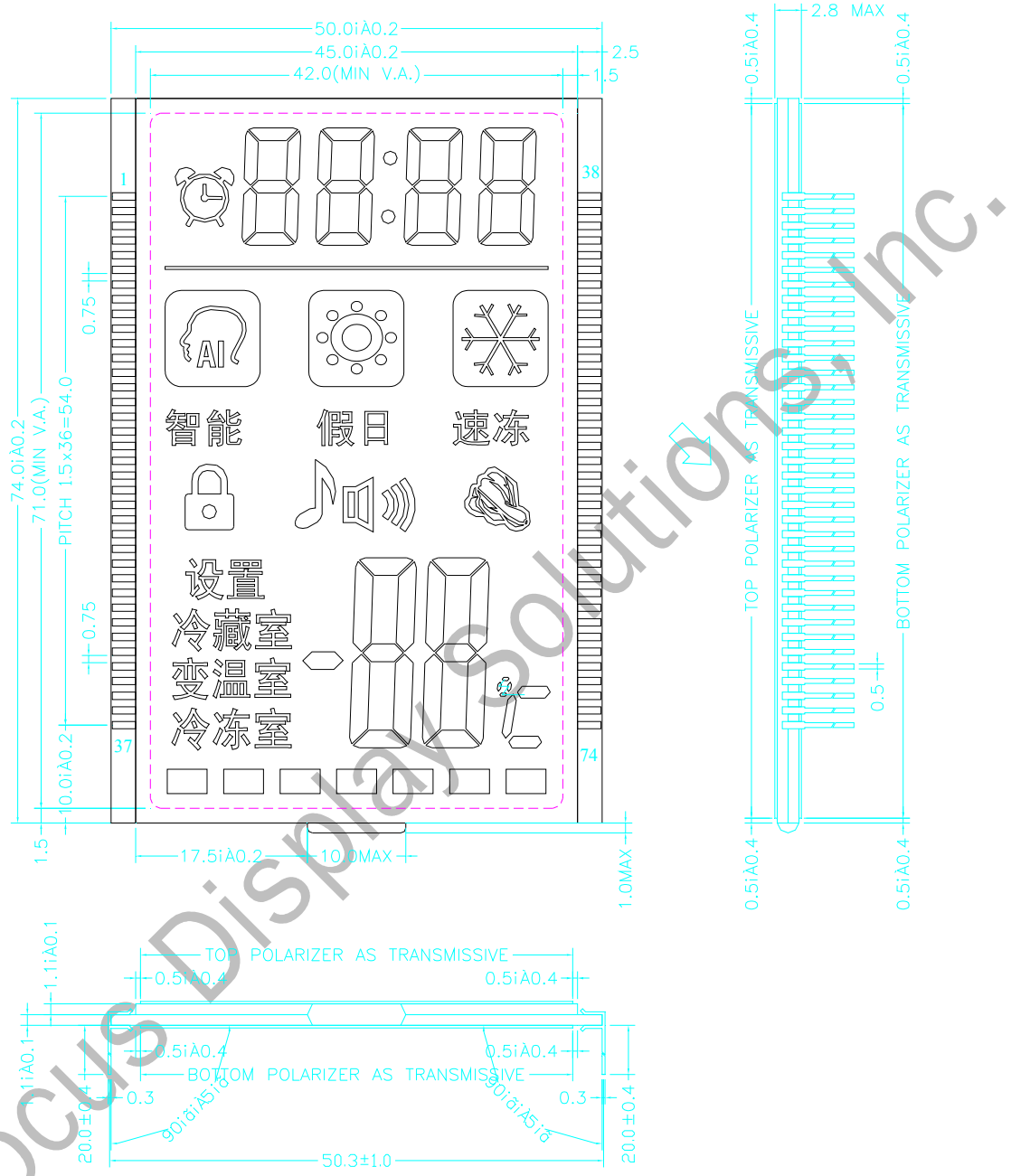
- High quality multicolor
- High contrast: above 100:1 (bright displaying avoids the insufficient black darkness in traditional LED)
- Wide view range(can identify active content distinctly at any view direction)
- Red,Green,Blue ,Yellow,Cyan,Purple,Black and Write 8 Color LCD
- No color filters, made by fast switching LCD and RGB LED backlight
- Wieldy interface
- Low cost and tooling charge
- Operating voltage: 5.0V
- Operation Temp: 0~70℃
- Sleep mode
- Display content: “8”+ logos
- Display type: FSLCD
- Display mode: Positive
- Polarizer type: Transmissive
- Driving method: Static

3 Mechanical Characteristics (Unit: mm):

External dimension:95.0*60.0 (±0.2)

View area : 70.0（MIN）*43.0（MIN）

4. LCM Dimension:



UNIT: mm

5. Environmental Characteristics:

Item	Symbol	Condition	Min	Max	Unit
Working Temperature	Twk		0	70	℃

Storage Temperature	Tstg		-20	80	℃

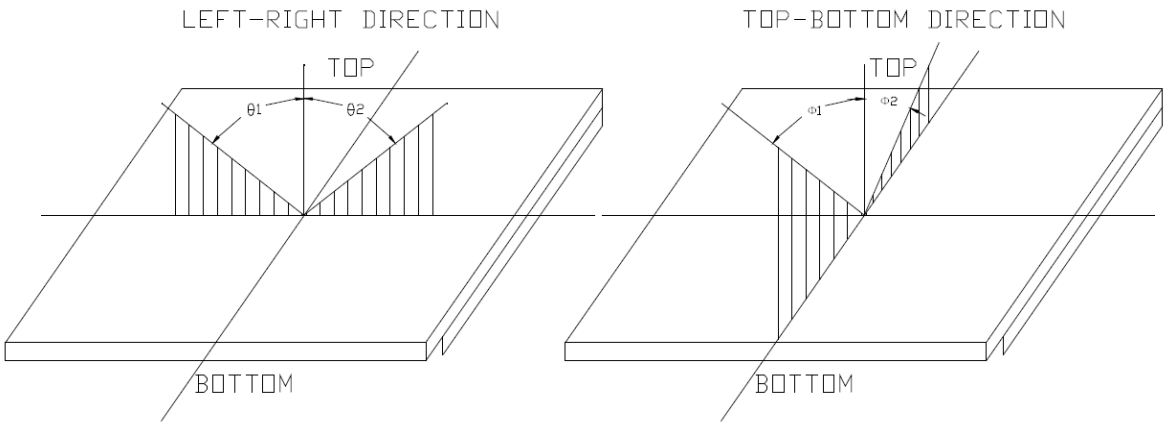
6. Electrical Characteristics:

Parameter	Symbol	Condition	Min	Type	Max	Unit
Supply Voltage for logic	Vdd-GND		4.75	5.0	5.25	V
Supply Current for logic	Idd	Vdd=5V	-	-	-	mA
Driving Voltage for LCM	Vdd-Vee		-9		-10	V
Driving Current for LCM	Iee		-	0.6	-	mA
Input Voltage H level	VIH		2.0			V
Input Voltage L level	VIL				0.8	V
Output Voltage H level	VOH	IOH=-10mA	2.4			V
Output Voltage L level	VOL	IOL=9.0mA			0.4	V
Normal mode1 Current	Inm1	Vdd=5V	50	80	90	mA
Sleeping mode Current	Islp1	Vdd=5V				mA

7. Optical Characteristics:

Parameter	Symbol	Condition	Min	Type	Max	UNIT	REF.
Contrast	CR	25℃，Vdd=5V, θ=0, Ø=0	--	50	--		(2)
Rise Time	Tr	25℃，Vdd=5V, θ=0, Ø=0	--	0.5	1	ms	(3)
Fall Time	Tf	25℃，Vdd=5V, θ=0, Ø=0	--	5	6	ms	(3)
Viewing Angle	θ1,θ2	25℃	--	50	--	DEG	(1)
	Ø1, Ø2		--	50	--		

(1)Definition of viewing Angle:

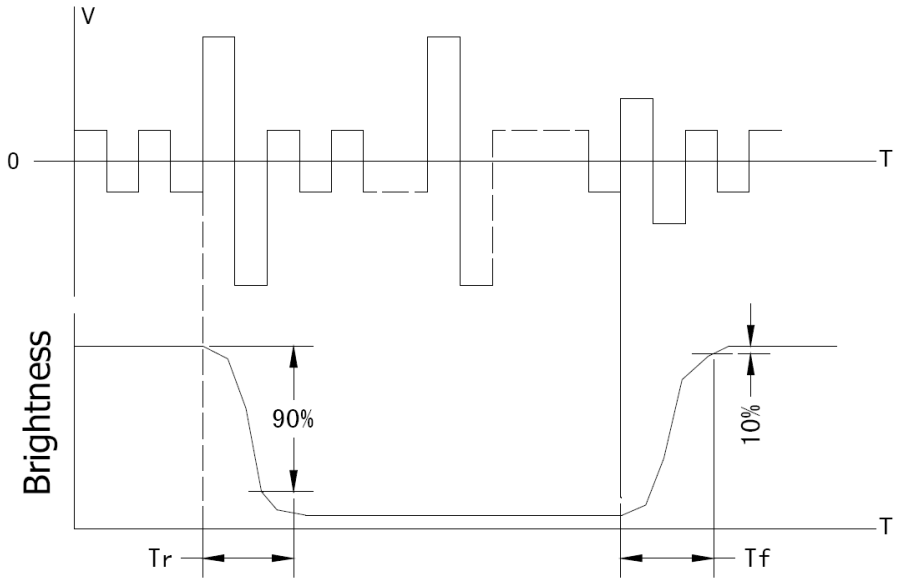


(2)Definition of Contrast Ratio:

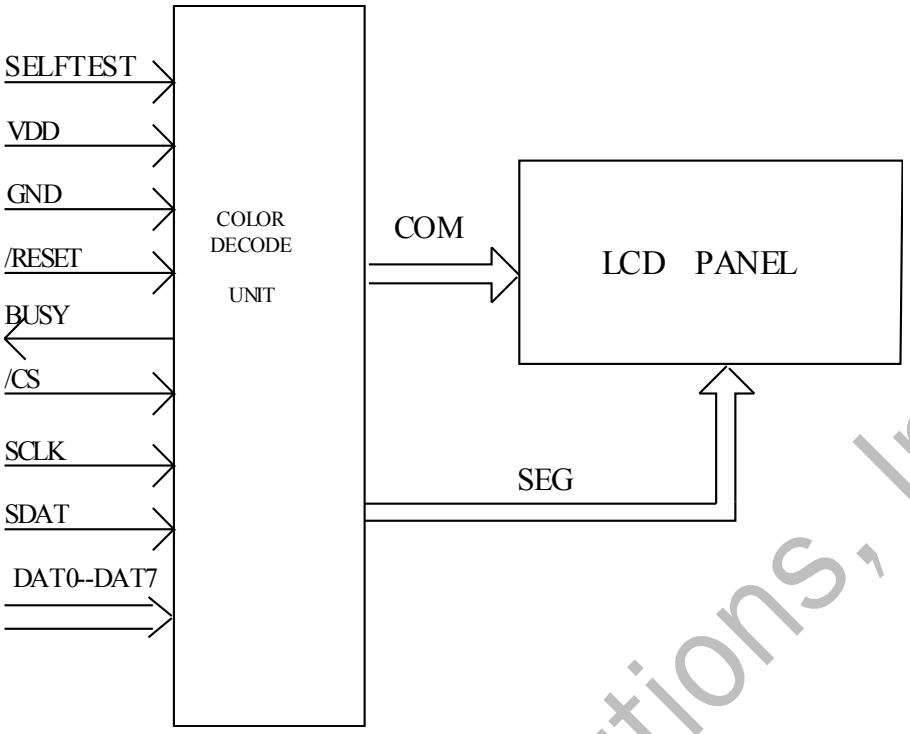
Contrast Ratio= Brightness of non-selected condition / Brightness of selected condition
Test condition: standard a light source

(3)Response Time:

Response time is measured as the shortest period of possible between the changes in state of an LCD segments as demonstrated below:



8. Block Diagram:



9. Pin Assignment:

Pin No.	Symbol	Function	Remark(Direction)
1	N/C	No connect	
2	SDAT	Serial Data input pin with serial transmission	MCU TO LCM
3	GND	Power Ground	
4	VDD	Power Supply	+5V
5	/RESET	LCM reset pin ,low plus LCM will reset	MCU TO LCM
6	BUSY	Busy Signal	LCM TO MCU
7	/CS	Chip Select , low is chip select	MCU TO LCM
8	SCLK	clock pulse pin with transmission	MCU TO LCM
9	DAT0	Parallel data0	MCU TO LCM
10	DAT1	Parallel data1	MCU TO LCM
11	DAT2	Parallel data2	MCU TO LCM
12	DAT3	Parallel data3	MCU TO LCM
13	DAT4	Parallel data4	MCU TO LCM
14	DAT5	Parallel data5	MCU TO LCM
15	DAT6	Parallel data6	MCU TO LCM
16	DAT7	Parallel data7	MCU TO LCM

Note:

1. LCM parallel interface pin：

4	3	5	6	7	8	9
VDD	VSS	/RESET	BUSY	/CS	SCLK	DAT0
10	11	12	13	14	15	16
DAT1	DAT2	DAT3	DAT4	DAT5	DAT6	DAT7

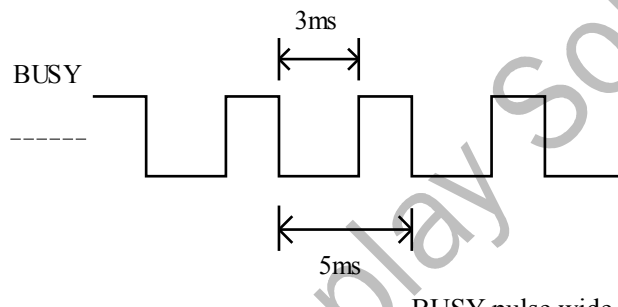
2. LCM serial interface pin:

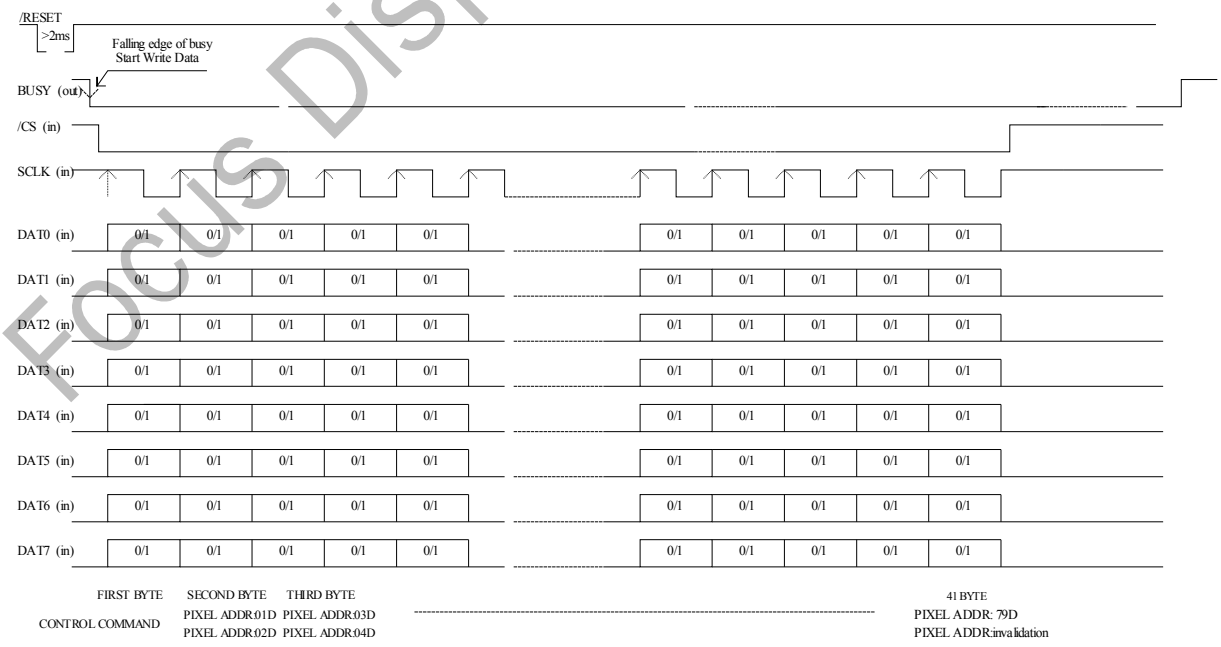
4	3	5	6	7	8	2
VDD	VSS	/RESET	BUSY	/CS	SCLK	SDAT

3. P/S PIN connects with VDD, working on **Serial** Mode (R9),
P/S PIN connects with VSS, working on **Parallel Mode** (RF7).

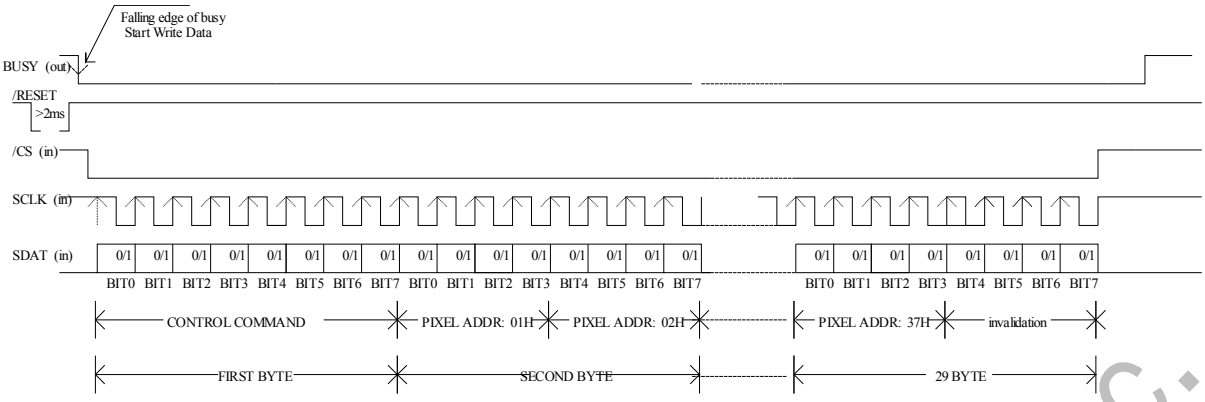
4. TEST PIN connects with VSS（R11）, working under the **SELFTEST** mode, for aging test of LCM,
When it work normally, connecting with VDD (R8).

10. Timing Diagrams:





LCM Parallel Communication Mode (One Frame Send to LCM)

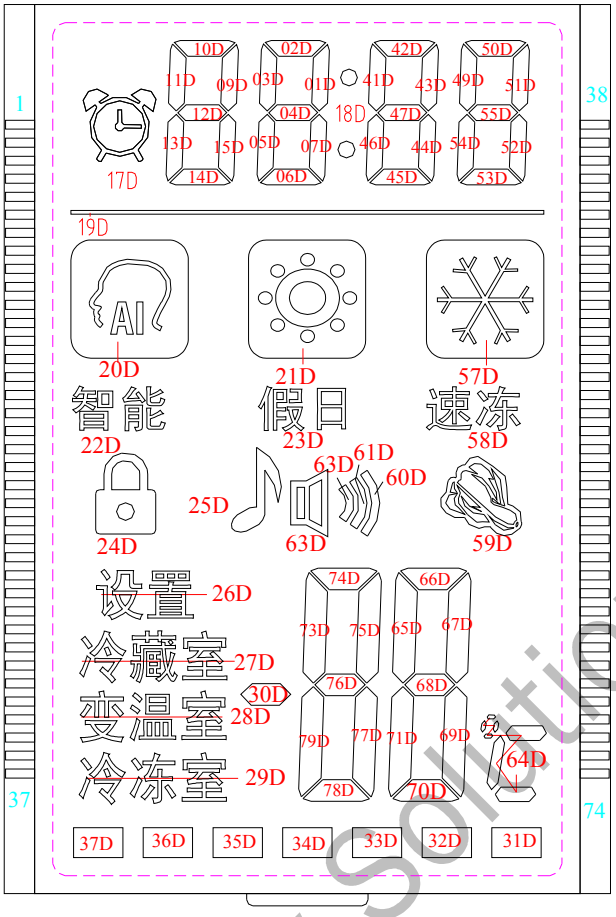


LCM Serial Communication Mode (One Frame Send to LCM)

Note:

1. When BUSY signal is high, can't write data; falling edge of BUSY, start write data. 2. Clock Rise Edge please send data to LCM, LCM will read data at clock Fall time.
3. If don't send data to LCM, please keep Sclk high.
4. One frame data must be transported within 3ms when BUSY on low-level, otherwise, screen sparkling occurred after data were transported.
5. Frame period: From the last transferred frame to the next frame need 100ms at least.

11. Pixel and Address map:



Each pixels address sees this figure.

12. Structure of Data (Control Command and Color command):

Total send 41 bytes data to LCM. First byte is Control Command, the others is Color Command.

One Color Command byte divided into two groups, high nibble [7:4] and low nibble [3:0].

High nibble [7:4] for one Pixel, and low nibble [3:0] for another pixel.

Send bytes		DATA							
		BIT 7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
1 byte	Control Command	Pleas see **Control Command**							
2 byte	Color	Pleas see Color Command				Pleas see Color Command			
	Pixel addr map	02D				01D			
3 byte	Color	Pleas see Color Command				Pleas see Color Command			
	Pixel addr map	04D				03D			
4 byte	Color	--				--			
	Pixel addr map	06D				05D			
5 byte	Color	--				--			
	Pixel addr map	0000B				07D			
6 byte	Color	--				--			
	Pixel addr map	10D				09D			
7 byte	Color	--				--			
	Pixel addr map	12D				11D			
8 byte	Color	--				--			
	Pixel addr map	14D				13D			
9 byte	Color	--				--			
	Pixel addr map	0000B				15D			
	Color	--				--			

10 byte	Pixel addr map	18D	17D
11 byte	Color	--	--
	Pixel addr map	20D	19D
12 byte	Color	--	--
	Pixel addr map	22D	21D
13 byte	Color	--	--
	Pixel addr map	24D	23D
14 byte	Color	--	--
	Pixel addr map	26D	25D
15 byte	Color	--	--
	Pixel addr map	28D	27D
16 byte	Color	--	--
	Pixel addr map	30D	29D
17 byte	Color	--	--
	Pixel addr map	32D	31D
18 byte	Color	--	--
	Pixel addr map	34D	33D
19 byte	Color	--	--
	Pixel addr map	36D	35D
20 byte	Color	--	--
	Pixel addr map	0000B	37D
21 byte	Color	--	--
	Pixel addr map	0000B	0000B
22 byte	Color	--	--
	Pixel addr map	42D	41D
23 byte	Color	--	--
	Pixel addr map	44D	43D
24 byte	Color	--	--
	Pixel addr map	46D	45D
25 byte	Color	--	--
	Pixel addr map	0000B	47D
26 byte	Color	--	--
	Pixel addr map	50D	49D
27 byte	Color	--	--
	Pixel addr map	52D	51D
28 byte	Color	--	--
	Pixel addr map	54D	53D
29 byte	Color	--	--
	Pixel addr map	0000B	55D
30 byte	Color	--	--
	Pixel addr map	58D	57D
31 byte	Color	--	--
	Pixel addr map	60D	59D
32 byte	Color	--	--
	Pixel addr map	62D	61D
33 byte	Color	--	--
	Pixel addr map	64D	63D
34 byte	Color	--	--
	Pixel addr map	66D	65D
35 byte	Color	--	--
	Pixel addr map	68D	67D
36 byte	Color	--	--
	Pixel addr map	70D	69D
37 byte	Color	--	--
	Pixel addr map	72D	71D
38 byte	Color	--	--
	Pixel addr map	74D	73D

39 byte	Color	--	--
	Pixel addr map	76D	75D
40 byte	Color	--	--
	Pixel addr map	78D	77D
41 byte	Color	--	--
	Pixel addr map	0000B	79D

NOTE:

-- means Pleas see Color Command.

0000B means high nibble [7:4] or low nibble [3:0] is 0H

13. Control Command:

Bit0	'0': must send '0'
Bit1	Sleep mode : '0': sleep off '1': sleep on
Bit2	Backlight Control Command: '0': backlight on 1': backlight off
Bit3	Bright Control: '0': normally bright '1': 1/2 bright
Bit4	Color Mode Control : '0':color mode0(normally color) '1': color mode1
Bit5	'1': must send '1'
Bit6	'1': must send '1'
Bit7	'1': must send '1'

14. Color Command:

One Color Command byte divided into two groups, high nibble [7:4] and low nibble [3:0].

high nibble [7:4] for one Pixel, and low nibble [3:0] for another pixel.

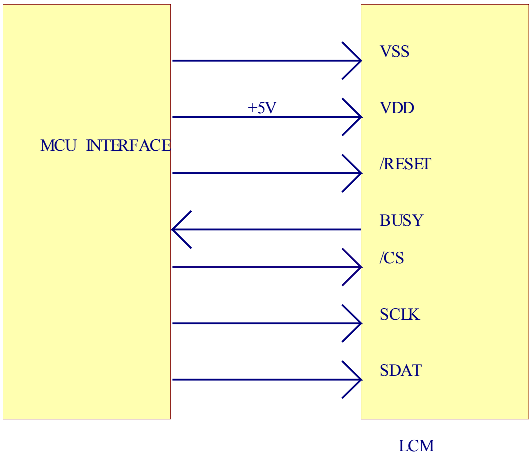
COLOR	BIT3	BIT2	BIT1	BIT0	HEX	Color Mode 0	Color Mode 1
	BIT7	BIT6	BIT5	BIT4			
White	0	0	0	0	0H		
Red	0	0	0	1	1H		
Green	0	0	1	0	2H		
Blue	0	0	1	1	3H		
Yellow	0	1	0	0	4H		
Cyan	0	1	0	1	5H		
Magenta	0	1	1	0	6H		
Black	0	1	1	1	7H		

15. Display Example:

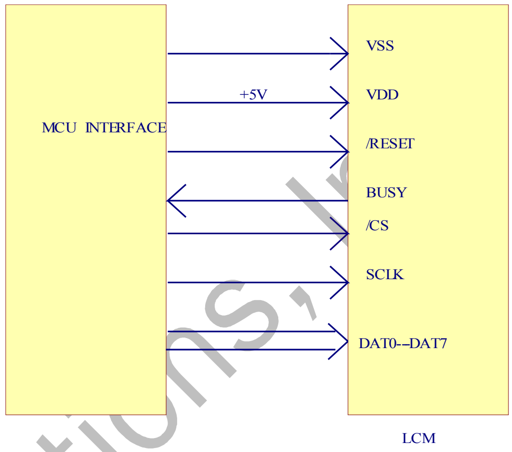
If we will control LCM in color mode1,we send F0h xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh to LCM by SCLK,total 41bytes. x meaning 0---7 color number.

If we will control pixel 03D is Yellow, we will send 00h xxh 04h xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh xxh to LCM by SCLK.

16. Application Circuits:



LCM Serial Interface



LCM Parallel Interface

17. Communication Code:

The following demo code can be used to implement communications to LCM.

I.Serial Interface mode:

```
/*****************************************************************************************************************/
#include<reg52.h>
#define uchar unsigned char
#define uint  unsigned int
/***PIN DEFINE*****************************************************************************************************/
sbit    busy =P1^0;
sbit    cs   =P1^1;
sbit    clk  =P1^2;
sbit    sdat =P1^3;
sbit    reset=P1^4;
/***0:WHITE 1:RED 2:GREEN 3:YELLOW 4:BLUE 5:MA 6:GRAY 7:BLACK/
/*****************0      1     2     3     4     5     6     7     8     9     10    11    12    13    14
                        15    16    17    18    19    20    21    22    23    24    25    26    27    28
                        29    30    31    32    33    34    35    36    37    38    39    40    */
/* Pixel Addr    CMD  02'01 04'03 06'05 08,07 10,09 12,11 14,13 16,15 18,17 20,19 22,21 24,23 26,25 28,27
                       30,29 32,31 34,33 36,35 38,37 40,39 42,41 44,43 46,45 48,47 50,49 52,51 54,53 56,55
                       58,57 60,59 62,61 64,63 66,65 68,67 70,69 72,71 74,73 76,75 78,77 80,79   */
uchar code S0[]={0x00,0x12, 0x34, 0x56, 0x71, 0x23, 0x15, 0x32, 0x54, 0x16, 0x32, 0x54, 0x36, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x11, 0x10, 0x11, 0x71, 0x11, 0x10, 0x11, 0x41,
                 0x11, 0x10, 0x11, 0x51, 0x11, 0x10, 0x11, 0x61, 0X65, 0X75, 0X57, 0X55};
uchar code S1[]={0x00,0x12, 0x34, 0x56, 0x71, 0x23, 0x61, 0x32, 0x54, 0x16, 0x32, 0x54, 0x16, 0X32, 0x54,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x00, 0x00, 0x10, 0x71, 0x00, 0x00, 0x10, 0x41,
                 0x00, 0x00, 0x10, 0x51, 0x00, 0x00, 0x10, 0x61, 0X65, 0X75, 0X57, 0X55};
uchar code S2[]={0x00,0x12, 0x34, 0x56, 0x71, 0x23, 0x55, 0x16, 0x32, 0x54, 0x16, 0x32, 0x54, 0X16, 0x32,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x11, 0x01, 0x11, 0x70, 0x11, 0x01, 0x11, 0x70,
                 0x11, 0x01, 0x11, 0x70, 0x11, 0x01, 0x11, 0x70, 0X65, 0X75, 0X57, 0X55};
uchar code S3[]={0x00,0x12, 0x34, 0x56, 0x71, 0x23, 0x45, 0x65, 0x21, 0x43, 0x65, 0x21, 0x43, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x01, 0x01, 0x11, 0x71, 0x01, 0x01, 0x11, 0x71,
                 0x01, 0x01, 0x11, 0x71, 0x01, 0x01, 0x11, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S4[]={0x00,0x12, 0x34, 0x56, 0x71, 0x23, 0x55, 0x16, 0x32, 0x54, 0x16, 0x32, 0x54, 0X16, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x00, 0x11, 0x10, 0x71, 0x00, 0x11, 0x10, 0x71,
                 0x00, 0x11, 0x10, 0x71, 0x00, 0x11, 0x10, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S5[]={0x00,0x12, 0x34, 0x56, 0x71, 0x23, 0x35, 0x54, 0x16, 0x32, 0x54, 0x16, 0x32, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x01, 0x11, 0x01, 0x71, 0x01, 0x11, 0x01, 0x71,
                 0x01, 0x11, 0x01, 0x71, 0x01, 0x11, 0x01, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S6[]={0x00,0x12, 0x34, 0x56, 0x71, 0x23, 0x25, 0x33, 0x54, 0x16, 0x32, 0x54, 0x36, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x11, 0x11, 0x01, 0x71, 0x11, 0x11, 0x01, 0x71,
                 0x11, 0x11, 0x01, 0x71, 0x11, 0x11, 0x01, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S7[]={0x00,0x12, 0x34, 0x56, 0x71, 0x23, 0x15, 0x32, 0x54, 0x16, 0x32, 0x54, 0x36, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x00, 0x00, 0x11, 0x71, 0x00, 0x00, 0x11, 0x71,
                 0x00, 0x00, 0x11, 0x71, 0x00, 0x00, 0x11, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S8[]={0x00,0x12, 0x34, 0x56, 0x71, 0x23, 0x45, 0x35, 0x55, 0x63, 0x33, 0x37, 0x37, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x11, 0x11, 0x11, 0x71, 0x11, 0x11, 0x11, 0x71,
                 0x11, 0x11, 0x11, 0x71, 0x11, 0x11, 0x11, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S9[]={0x00,0x12, 0x34, 0x56, 0x71, 0x23, 0x45, 0x35, 0x55, 0x63, 0x33, 0x37, 0x37, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x01, 0x11, 0x11, 0x71, 0x01, 0x11, 0x11, 0x71,
                 0x01, 0x11, 0x11, 0x71, 0x01, 0x11, 0x11, 0x71, 0X65, 0X75, 0X57, 0X55};
/*****************************************************************************************************************/
void    delaytime(uint x)
{
        uchar  i,j;
        while(x--)
        {
        for(j=0;j<50;j++)
         {
          for(i=0;i<125;i++)
          {;}
         }
        }
}

void    delay()
{       uchar  i;
        for(i=0;i<8;i++)
        {;}
```

```
}
void    sendlcd_s(uchar temp)
{       uchar m;
        for(m=0;m<8;m++)
        {
        clk=1;
        if(temp&0x01)
        sdat=1;
        else
        sdat=0;
        temp=temp>>1;
//      delay();
        clk=0;
//      delay();
        }
}
void    se_dat(uchar Array[])
{
        uchar j;
        clk=1;
        while(busy==0)  //=1,next    /_____/————/_____/
        {;}
        while(busy==1)  //=0,next    falling edge of Busy, Start write Data
        {;}
        cs=0;
        for(j=0;j<41;j++)
        {
        sendlcd_s(Array[j]);
        }
        cs=1;
        clk=1;
}
void    main()
{
        uchar   nu;
        //initialize   color lcd
        sdat=1;
        clk=1;
        cs=1;
        reset=0;
        nu=20;
        delaytime(nu);
        reset=1;
        while(1)
        {
        nu=5;
        se_dat(S0);delaytime(nu);
        nu=5;
        se_dat(S1);delaytime(nu);
        nu=5;
        se_dat(S2);delaytime(nu);
        nu=5;
        se_dat(S3);delaytime(nu);
        nu=10;
        se_dat(S4);delaytime(nu);
        nu=10;
        se_dat(S5);delaytime(nu);
        nu=10;
        se_dat(S6);delaytime(nu);
        nu=15;
        se_dat(S7);delaytime(nu);
        nu=35;
        se_dat(S8);delaytime(nu);
        nu=45;
        se_dat(S9);delaytime(nu);
        }
```

II. Parallel Interface mode

```
/*********************************************************************************************************/
#include<reg52.h>
#define uchar unsigned char
#define uint  unsigned int
/***PIN DEFINE*******************************************************************************************/
#define Pdata P2
sbit    busy    =P1^0;
sbit    cs      =P1^1;
sbit    clk     =P1^2;
sbit    sdat    =P1^3;
sbit    reset   =P1^4;
/***0:WHITE 1:RED 2:GREEN 3:YELLOW 4:BLUE 5:MA 6:GRAY 7:BLACK/
/*****************0      1     2     3     4     5     6     7     8     9     10    11    12    13    14
                        15    16    17    18    19    20    21    22    23    24    25    26    27    28
                        29    30    31    32    33    34    35    36    37    38    39    40              */
/*****Color ADDR CMD   02'01 04'03 06'05 08,07 10,09 12,11 14,13 16,15 18,17 20,19 22,21 24,23 26,25 28,27
                        30,29 32,31 34,33 36,35 38,37 40,39 42,41 44,43 46,45 48,47 50,49 52,51 54,53 56,55
                        58,57 60,59 62,61 64,63 66,65 68,67 70,69 72,71 74,73 76,75 78,77 80,79          */
uchar code S0[]={0x00, 0x12, 0x34, 0x56, 0x71, 0x23, 0x15, 0x32, 0x54, 0x16, 0x32, 0x54, 0x36, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x11, 0x10, 0x11, 0x71, 0x11, 0x10, 0x11, 0x41,
                 0x11, 0x10, 0x11, 0x51, 0x11, 0x10,  0x11, 0x61, 0X65, 0X75, 0X57, 0X55};
uchar code S1[]={0x00, 0x12, 0x34, 0x56, 0x71, 0x23, 0x61, 0x32, 0x54, 0x16, 0x32, 0x54, 0x16, 0X32, 0x54,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x00, 0x00, 0x10, 0x71, 0x00, 0x00, 0x10, 0x41,
                 0x00, 0x00, 0x10, 0x51, 0x00, 0x00,  0x10, 0x61, 0X65, 0X75, 0X57, 0X55};
uchar code S2[]={0x00, 0x12, 0x34, 0x56, 0x71, 0x23, 0x55, 0x16, 0x32, 0x54, 0x16, 0x32, 0x54, 0X16, 0x32,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x11, 0x01, 0x11, 0x70, 0x11, 0x01, 0x11, 0x70,
                 0x11, 0x01, 0x11, 0x70, 0x11, 0x01,  0x11, 0x70, 0X65, 0X75, 0X57, 0X55};
uchar code S3[]={0x00, 0x12, 0x34, 0x56, 0x71, 0x23, 0x45, 0x65, 0x21, 0x43, 0x65, 0x21, 0x43, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x01, 0x01, 0x11, 0x71, 0x01, 0x01, 0x11, 0x71,
                 0x01, 0x01, 0x11, 0x71, 0x01, 0x01,  0x11, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S4[]={0x00, 0x12, 0x34, 0x56, 0x71, 0x23, 0x55, 0x16, 0x32, 0x54, 0x16, 0x32, 0x54, 0X16, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x00, 0x11, 0x10, 0x71, 0x00, 0x11, 0x10, 0x71,
                 0x00, 0x11, 0x10, 0x71, 0x00, 0x11,  0x10, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S5[]={0x00, 0x12, 0x34, 0x56, 0x71, 0x23, 0x35, 0x54, 0x16, 0x32, 0x54, 0x16, 0x32, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x01, 0x11, 0x01, 0x71, 0x01, 0x11, 0x01, 0x71,
                 0x01, 0x11, 0x01, 0x71, 0x01, 0x11,  0x01, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S6[]={0x00, 0x12, 0x34, 0x56, 0x71, 0x23, 0x25, 0x33, 0x54, 0x16, 0x32, 0x54, 0x36, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x11, 0x11, 0x01, 0x71, 0x11, 0x11, 0x01, 0x71,
                 0x11, 0x11, 0x01, 0x71, 0x11, 0x11,  0x01, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S7[]={0x00, 0x12, 0x34, 0x56, 0x71, 0x23, 0x15, 0x32, 0x54, 0x16, 0x32, 0x54, 0x36, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x00, 0x00, 0x11, 0x71, 0x00, 0x00, 0x11, 0x71,
                 0x00, 0x00, 0x11, 0x71, 0x00, 0x00,  0x11, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S8[]={0x00, 0x12, 0x34, 0x56, 0x71, 0x23, 0x45, 0x35, 0x55, 0x63, 0x33, 0x37, 0x37, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x11, 0x11, 0x11, 0x71, 0x11, 0x11, 0x11, 0x71,
                 0x11, 0x11, 0x11, 0x71, 0x11, 0x11,  0x11, 0x71, 0X65, 0X75, 0X57, 0X55};
uchar code S9[]={0x00, 0x12, 0x34, 0x56, 0x71, 0x23, 0x45, 0x35, 0x55, 0x63, 0x33, 0x37, 0x37, 0X62, 0x27,
                 0X22, 0x27, 0x67, 0x11, 0x77, 0x77, 0x01, 0x11, 0x11, 0x71, 0x01, 0x11, 0x11, 0x71,
                 0x01, 0x11, 0x11, 0x71, 0x01, 0x11,  0x11, 0x71, 0X65, 0X75, 0X57, 0X55};
/*********************************************************************************************************/
void delaytime(uchar f)
{
        uchar   i,j;
        while(f--)
        {
          for(j=0;j<50;j++)
          {
            for(i=0;i<125;i++)
            {;}
          }
        }
}

void delay()
{
        uchar    i;
```

```
        for(i=0;i<5;i++)
        {;}
}

void se_dat(uchar Array[])
{
        uchar j;
        while(busy==0)          //=1, next      /_____/————/____/
        {;}
        while(busy==1)          //=0, next     falling edge of Busy, Start write Data
        {;}
        clk=1;
        cs=0;
        delay();
        for(j=0;j<41;j++)
        {
            clk=1;
            Pdata=Array[j];
            clk=0;
            delay();
        }
        cs=1;
        clk=1;
}

main()
{
        uchar   nu;
        P1      =0XFF;
        Pdata   =0XFF;
        reset   =0;
        nu      =1;
        delaytime(nu);
        P1      =0XFF;
        Pdata   =0XFF;
        while(1)
        {
        nu=20;
        se_dat(S0);delaytime(nu);
        nu=30;
        se_dat(S1);delaytime(nu);
        nu=40;
        se_dat(S2);delaytime(nu);
        nu=50;
        se_dat(S3);delaytime(nu);
        nu=60;
        se_dat(S4);delaytime(nu);
        nu=70;
        se_dat(S5);delaytime(nu);
        nu=80;
        se_dat(S6);delaytime(nu);
        nu=90;
        se_dat(S7);delaytime(nu);
        nu=100;
        se_dat(S8);delaytime(nu);
        nu=120;
        se_dat(S9);delaytime(nu);
    }
}
```

18. FS-LCM Capability Test

TEST ITEM		SYMBOL	MIN.	TYP.	MAX.	UNIT	RESULT
Driving Voltage		Vop(25℃)	4.4	5.0	5.5	V	Pass
Driving Frequency		Ff	135	180	230	Hz	Pass
LCD Consumption Current		Is	…..	…..	1	μa/cm ²	Pass
LCD Consumption Current		Ic	…..	…..	70	mA	Pass
Response Time	Raise Time	Ton(25℃)	…..	…..	6	ms	Pass
	Fall Time	Toff((25℃)	…..	…..	10	ms	Pass
Viewing Angle	Top&Bottom	φ1, φ2	-70	…..	70	°	Pass
	L&R	θ1, θ2	-70	…..	70	°	Pass
Luminance		Bright	100	102		cd/ m ²	Pass
Chroma		Color	X=0.20 Y=0.20	X=0.22 Y=0.22			Pass
Contrast Ratio		Cr		1:100	…..		Pass
Working Temp.		Tstg	0		70	℃	Pass
Remark		0℃~-20 ℃, portion color will be getting black and white display, detailed explain please see the attachment.					
Storage Temp.		Top	-20		80	℃	Pass
Static Test		LCD		LCM(PIN Connector)		LCM(Metal Connector)	
		+	-	+	-	+	-
Contact Discharge		5000V	5000V	2000V	2000V	5000V	5000V
Air Discharge		8000V	8000V	5000V	5000V	8000V	8000V

19. Reliability Test

Conducted by MIL-STD-105E							
MIJ：0.65 MIN:2.5 Sampling Method : MIJ：0.65 MIN:2.5					JUGEMENT		
					Result		
No.	Test Item,	Criterion & Attention		Method	CRI	MAJ	MIN
4	**High Temperature Operation**	65℃, 96hrs	Appearance should not be changed, contrast's discrepancy should not be ±20% with initial value. Total current consumption should be below double of initial value.	Record the data and compare the performance.		.	
	Low Temperature Operation	-25℃, 96hrs					
	Temperature Operation	40℃，90%,96 hrs	Appearance should not be changed, contrast's discrepancy should not be ±20% with initial value. Total current consumption should be below double of initial value.				
	High Temperature Storage	70℃，96hrs					
	Low Temperature Storage	-25℃ 96hrs					
	Thermal Shock	-25℃→30℃→25℃→70℃ 5（min）30（min）5（min）30（min）5 Cycles，55-60%RH					
	Vibration	10-55-10HZ amplitude:1.5mm 2hrs for each direction	Appearance and reliability should not be changed; total current Consumption should be below double of initial value.				

20. Precaution for Using LCM:

1. Liquid Crystal Display (LCD)

LCD is made up of glass, organic sealant, organic fluid, and polymer based polarizers. The

following precautions should be taken when handing,
(1). Keep the temperature within range of use and storage. Excessive temperature and humidity could cause polarization degradation, polarizer peel off or bubble.
(2). Do not contact the exposed polarizers with anything harder than an HB pencil lead. To clean dust off the display surface. Wipe gently with cotton. Chamois or other soft material soaked in petroleum benzene.
(3). Wipe off saliva or water drops immediately. Contact with water over a long period of time may cause polarizer deformation or color fading, while an active LCD with water condensation on its surface will cause corrosion of ITO electrodes.
(4). Glass can be easily chipped or cracked from rough handing. Especially at corners and edges.

2. Liquid Crystal Display Modules

2.1 Mechanical Considerations

LCM are assembled and adjusted with a high degree of precision. Avoid excessive shocks and do not make any alterations or modifications. The following should be noted.
(1). Do not tamper in any way with the tabs on the tabs on the metal frame.
(2). Do not modify the PCB by drilling extra holes, changing its outline, moving its components or modifying its pattern.
(3). Do not touch the elastomer connector; especially insert a backlight panel (for example, EL).
(4). When mounting a LCM makes sure that the PCB is not under any tress such as bending or twisting.
Elastomer contacts are very delicate and missing pixels could result from slight dislocation of any of the elements.
(5). Avoid pressing on the metal bezel, otherwise the elastomer connector could be deformed and lose contact, resulting in missing pixels.

2.2. Static Electricity

LCM contains CMOS LSI's and the same precaution for such devices should apply, namely
(1). The operator should be grounded whenever he/she comes into contact with the module. Never touch any of the conductive parts such as the LSI pads, the copper leads on the PCB and the interface terminals with any parts of the human body.
(2). The modules should be kept in antistatic bags or other containers resistant to static for storage.
(3). Only properly grounded soldering irons should be used.
(4). If an electric screwdriver is used, it should be well grounded and shielded from commutator sparks.
(5). The normal static prevention measures should be observed for work clothes and working benches; for the latter conductive (rubber) mat is recommended.
(6). Since dry air is inductive to statics, a relative humidity of 50-60% is recommended.

2.3. Soldering

(1). Solder only to the I/O terminals.
(2). Use only soldering irons with proper grounding and no leakage.

(3). Soldering temperature: 280 ℃±10℃

(4). Soldering time: 3 to 4 sec.
(5). Use eutectic solder with resin flux fill.
(6). If flux is used, the LCD surface should be covered to avoid flux spatters. Flux residue should be
removed after wards.

2.4. Operation

(1). The viewing angle can be adjusted by varying the LCD driving voltage V0.
(2). Driving voltage should be kept within specified range; excess voltage shortens display life.
(3). Response time increases with decrease in temperature.
(4). Display may turn black or dark blue at temperatures above its operational range; this is (however not pressing on the viewing area) may cause the segments to appear “fractured”.
(5). Mechanical disturbance during operation (such as pressing on the viewing area) nay cause the segments to appear “fractured”.

2.5. Storage

If any fluid leaks out of a damaged glass cell, wash off any human part that comes into contact with soap and water. Never swallow the fluid. The toxicity is extremely low but caution should be exercised
at all the time.