SPECIFICATIONS FOR
LIQUID CRYSTAL DISPLAY MODULE

CUSTOMER	
CUSTOMER Part No:	
MODULE No:	**ACM041BWWB-H-U**
DGW VERSION:	**REV.0**

APPROVED SIGNATURE

YUDU AMSON ELECTRONICS CO.,LTD.
TEL : 86-797-6330063
FAX : 86-797-6330055

APPROVED BY	CHECKED BY	PREPARED BY	ISSUED DATE:
	HSU	GILI	Oct.28.2022

Revision Record

No.	Date	Model No.	Version	Remarks
1	Oct.28.2022	ACM041BWWB-H-U	REV.0	SPEC RoHs-Compliant

Contents

Module Classification Information

A	CM	041	B	W	W	B	-	H		-	U	
1	2	3	4	5	6	7		8	9		10	11

1	**YUDU AMSON Displays, Inc.**		
2	**Product type：OLED COG**		
3	**Display Type：Graphic 128 x 64 Dots**		
4	**Serials No.**		
5	**LCD Type：**	N → TN/ Positive / Transflective / Reflective H → TN/ Positive/ Transmissive S → STN/ Gray/ Positive / Transflective / Reflective U → STN/ Y-G/ Positive / Transflective / Reflective F → FSTN/ Positive / Transflective / Reflective K → FSTN/ Positive/ Transmissive	M → TN/ Negative / Transmissive B → STN/ Blue/ Negative / Transmissive Z → STN/ Gray/ Positive/ Transmissive T → STN/ Y-G/ Positive/ Transmissive W → FSTN/ Negative / Transmissive W → OLED & Black
6	**Backlight Type：**	L→ LED / Yellow-Green M→ LED / Amber N→ LED / Red O→ LED / Orange	G→ LED / Green / Guide W→ LED / White / Guide F→ LED / Blue / Guide *→ No BackLight / Reflective
7	**LCD View direction /**	B→ 6:00 View T→ 12:00 View	
8	**LCD Temp. Range：**	H → Wide Temp., 5V , Single Supply Voltage W → Wide Temp., 3.3V, Single Supply Voltage	
9	**Frame Material**	→ NO FRAME	B→ BLACK S→ SILVER
10	**Series Number**	X→ A~Z Number	
11	**Series Number**	X→ --- or 01~99 Number	

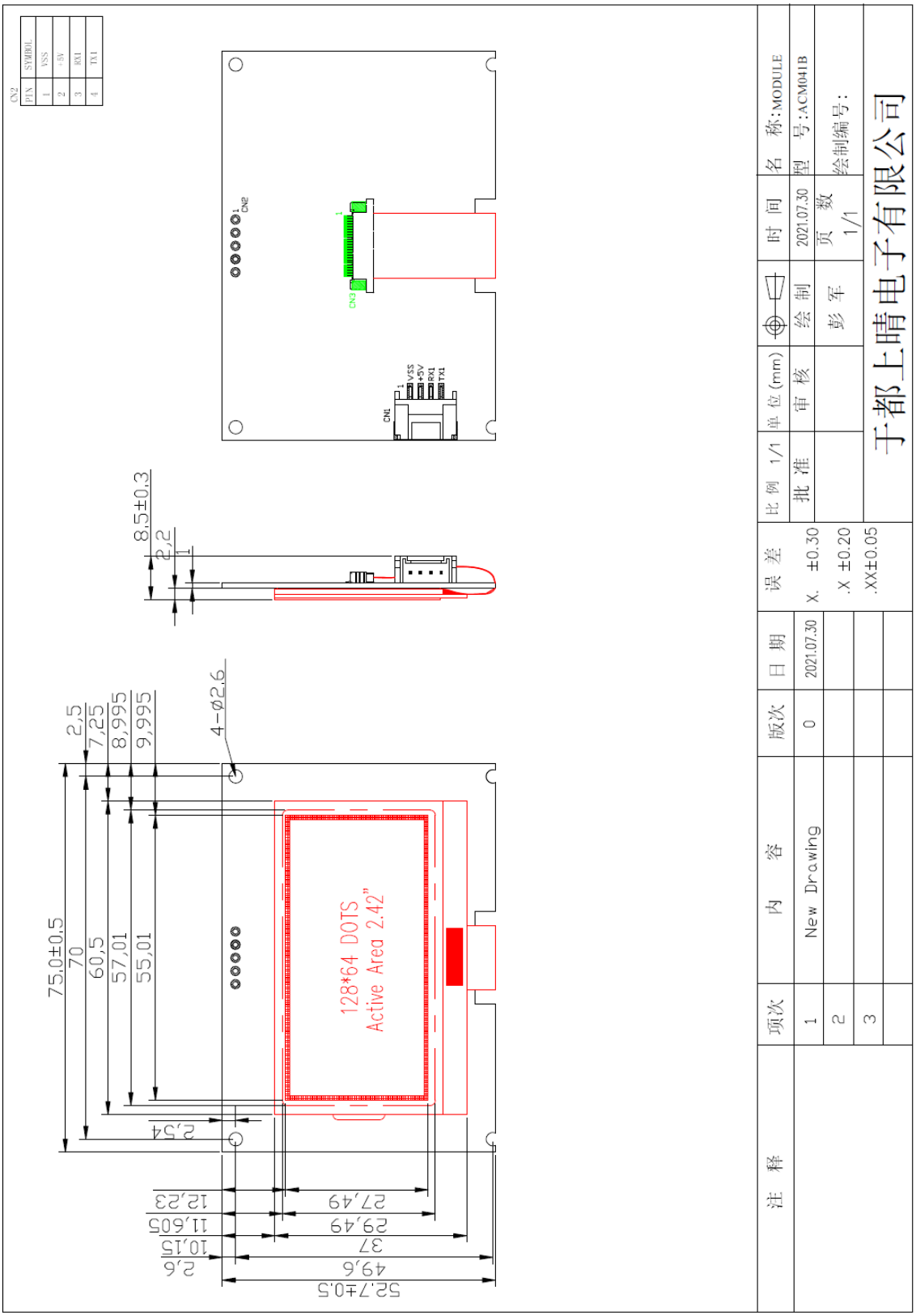
1. Precautions in use of OLED Modules

(1) Avoid applying excessive shocks to the module or making any alterations or modifications to it.
(2) Don't make extra holes on the printed circuit board, modify its shape or change the components of OLED module.
(3) Don't disassemble the LCM.
(4) Don't operate it above the absolute maximum rating.
(5) Don't drop, bend or twist LCM.
(6) Soldering: only to the I/O terminals.
(7) Storage: please storage in anti-static electricity container and clean environment.
Don't touch the elastomers connecter, especially insert a backlight panel (EL or CCFL)
(8) LCM technological conditions RoHS

2. General Specification

Item	Dimension	Unit
Dot arrangement	128 x 64	Dots
Module dimension	75.0 x 52.7 x 8.8 Max	mm
View area	57.01 x 29.9	mm
Active area	55.01 x 27.49	mm
Dot size	0.40 x 0.40	mm
Dot pitch	0.43 x 0.43	mm
Display mode	Passive Matrix	
Duty Ratio	1/64Duty	
View direction	6 o'clock	
Display color	OLED White & Black	
Controller Interface	SSD1309	
Controller Interface	M031TD2AE UART 115200	
Built-in	With DC/DC Converter	

3. Mechanical Drawing & Interface



3.1 Interface Pin Description

Pin No.	Pin Out	Level	Description
1	TX1	H/L	Serial Transmit Signal
2	RX1	H/L	Serial Receive Signal
3	+5V	+5V	Power Supply Voltage
4	VSS	0V	Power Supply Ground.

4. Absolute Maximum Ratings

4.1 Electrical Absolute Ratings

Item	Symbol	Min.	Max.	Unit	Note
Power Supply for Logic	V_{DD}-V_{SS}	-0.3	7.0	Volt	
Power Supply for OLED	V_{CC}-V_{SS}	-0.3	15.0	Volt	
Input Voltage	V_I	-0.3	V_{DD}	Volt	
Life Time (60 cd/m^2)	T_a = 25°C 50% RH	50,000	---	Hour	Note 1

4.2 Environmental Absolute Maximum Ratings

Item	Wide Temperature			
	Operating		Storage	
	Min,	Max.	Min,	Max.
Ambient Temperature	-40℃	+75℃	-40℃	+80℃
Humidity (without condensation)	Note 2,3		Note 2,3	

Note 1: Software configuration follows. End of lifetime is specified as 50% of initial brightness reached. The average operating lifetime at room temperature is estimated by the accelerated operation at high temperature conditions.

Note 2: Background color changes slightly depending on ambient temperature. This phenomenon is reversible.

Note 3: The defined temperature ranges do not include the polarizer. The maximum withstood temperature of the polarizer should be 80°C.

5. Electrical Characteristics

5.1 DC Characteristics for logic and OLED Ta=25℃,VSS=0V

Item	Symbol	Condition	Min.	Typ	Max.	Unit	note
Power Supply for Logic	V_{DD}-V_{SS}	-	4.5	5.0	5.0	Volt	
Power Supply for OLED	V_{DD}-V_{SS}	-	2.7	3.3	3.6	Volt	
Power Supply for OLED	V_{CC}-V_{SS}	-	12.5	13.0	13.5	Volt	Note 5
Input Voltage	V_{IL}	L level	0	-	$0.2V_{DD}$	Volt	
	V_{IH}	H level	$0.8V_{DD}$	-	V_{DD}	Volt	
Onput Voltage	V_{OL}	L level	0	-	$0.1V_{DD}$		
	V_{OH}	H level	$0.9V_{DD}$	-	V_{DD}		
Operating Current for Vpp	I_{PP}	Note 6	-	29.7	35.8	mA	
		Note 7	-	31.6	41.7		
		Note 8	-	39.6	54.9		
Power Supply Current for OLED	I_{DD}	V_{DD} =3.3V Vpp =13.0V	-	39.9	55.4	mA	

Note 5: Brightness (Lbr) and Supply Voltage for Display (Vpp) are subject to the change of the panel characteristics and the customer's request.

Note 6: V_{DD} = 3.3V, Vpp =13.0V generated by internal DC/DC converter, 30% Display Area Turn on.

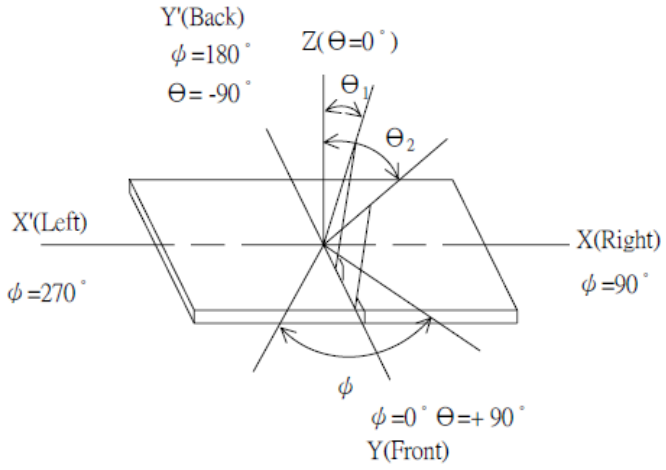
Note 7: V_{DD} = 3.3V, Vpp =13.0V generated by internal DC/DC converter, 50% Display Area Turn on. (POR)

Note 8: V_{DD} = 3.3V, Vpp =13.0V generated by internal DC/DC converter, 100% Display Area Turn on.

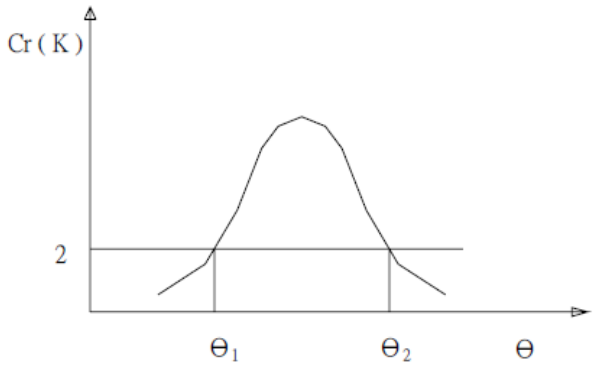
6. Optical Characteristics

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	note
Viewing Angle Range	Θ2－Θ1	Ta＝25℃	80	-	-	deg.	1,2
	ϕ		80	-	-	deg.	
Contrast Ratio	CR	Ta＝25℃	CR >10000:1			-	3
Response Time(rise)	Tr	Ta＝25℃	-	10	-	us	4
Response Time(fall)	Tf	Ta＝25℃	-	10	-	us	4
Brightness	L	Check Board Brightness	60	80	-	cd/m^2	
Peak Emission Wavelength	C.I.E (White)		X=0.25 Y=0.27	X=0.29 Y=0.31	X=0.33 Y=0.35	nm	

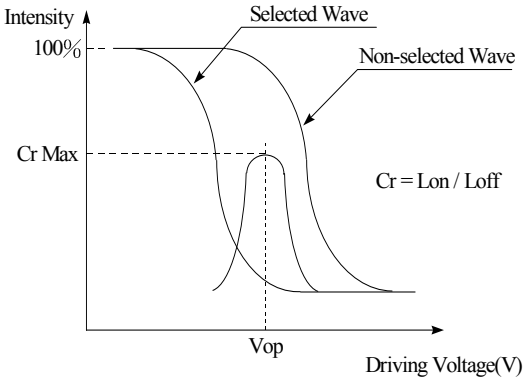
Note1: Definition of angle θ and Φ



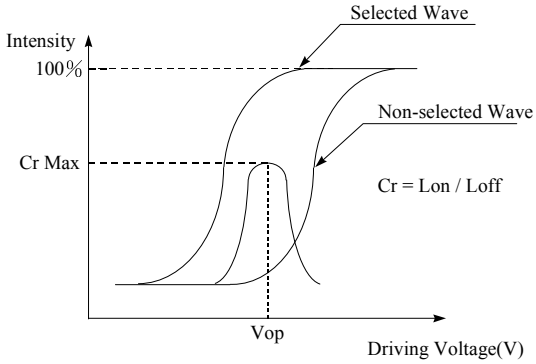
Note2: Definition of viewing angle θ



Note3: Definition of contrast CR

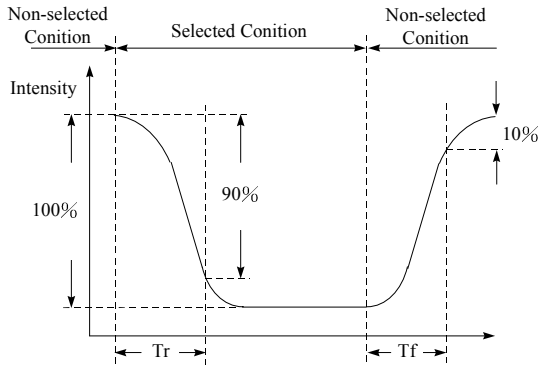


[positive type]

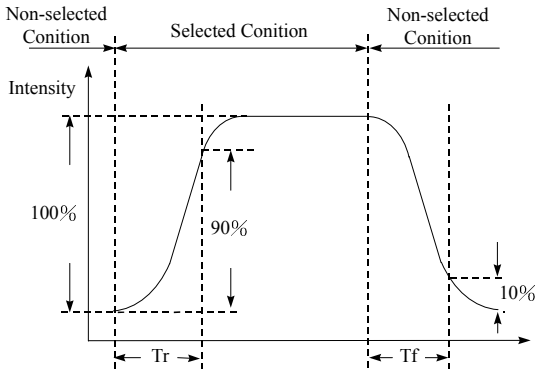


[Negative type]

Note4: Definition of Response Time (Tr, Tf)



[positive type]

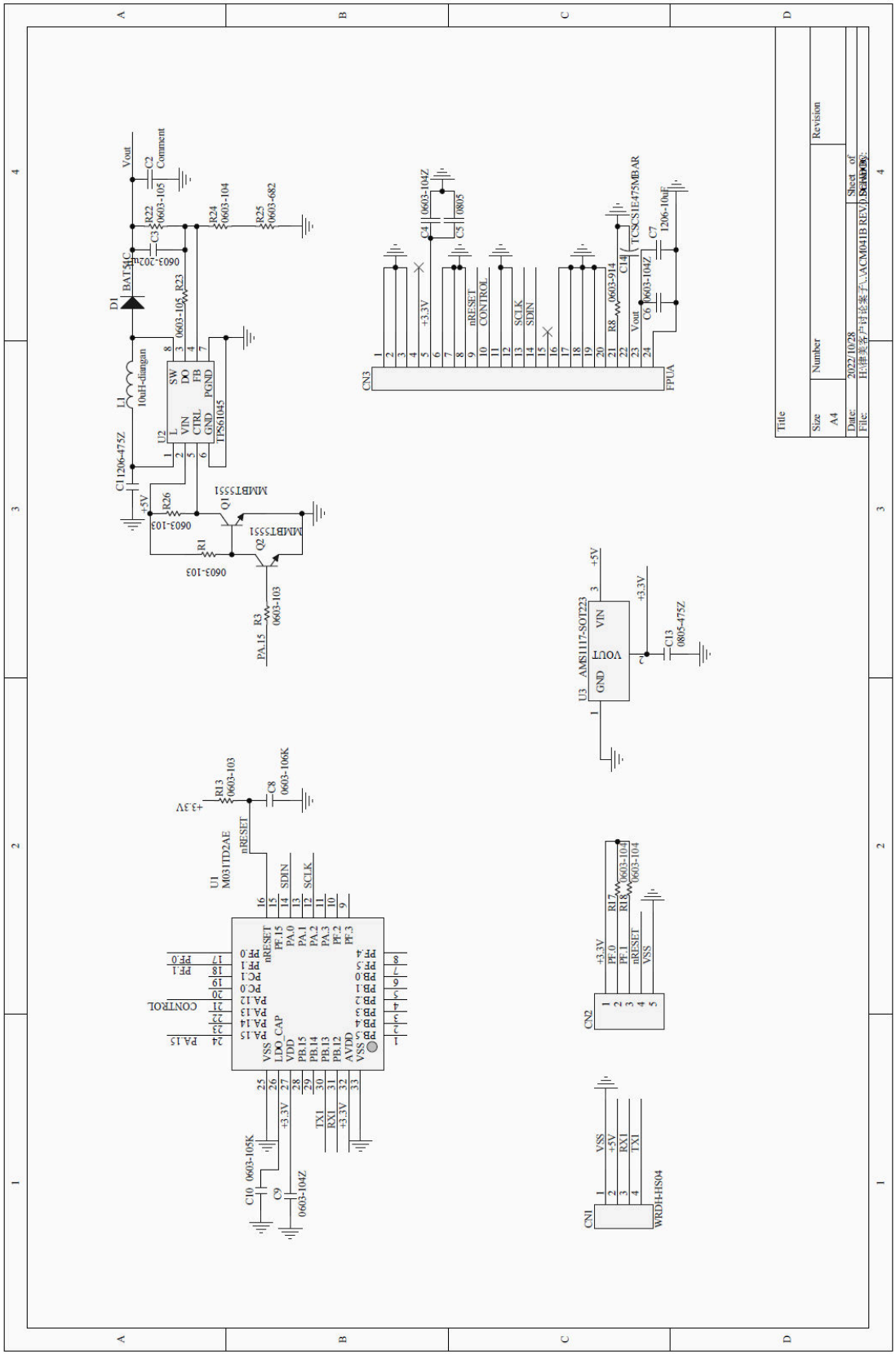


[Negative type]

Conditions: Operating Voltage : Vop
Frequency : 64 Hz

Viewing Angle(θ, φ): 0° , 0°Frame
Driving Wave form : 1/N duty, 1/a bias

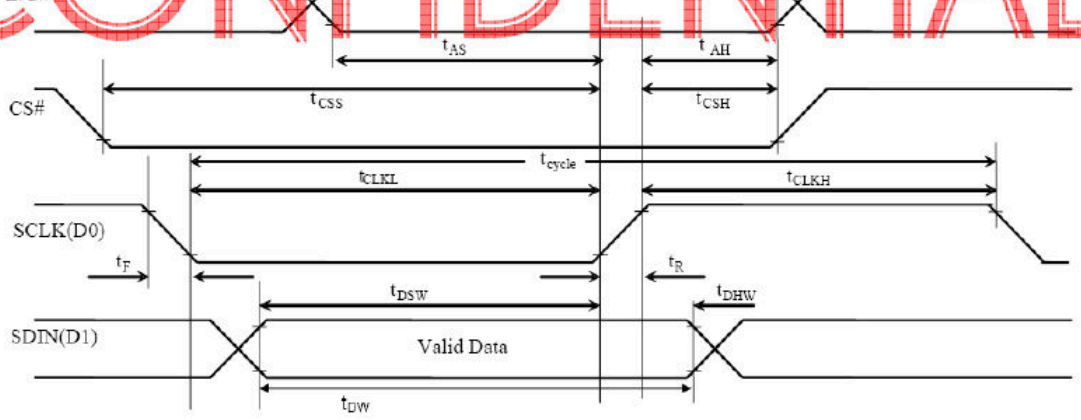
7. Block Diagram

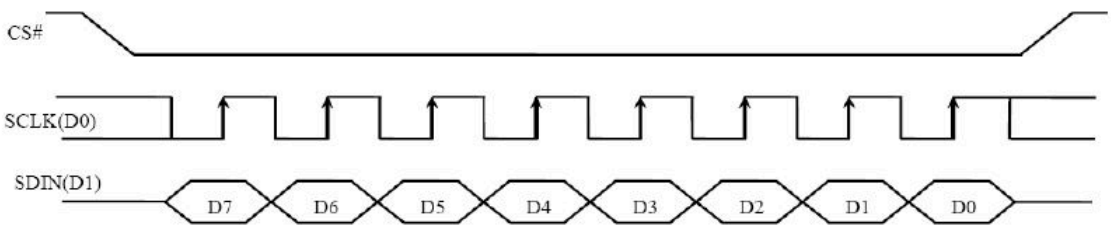


8. AC Characteristics

Symbol	Description	Min	Max	Unit
t_{cycle}	Clock Cycle Time	100	-	ns
t_{AS}	Address Setup Time	15	-	ns
t_{AH}	Address Hold Time	15	-	ns
t_{CSS}	Chip Select Setup Time	20	-	ns
t_{CSH}	Chip Select Hold Time	50	-	ns
t_{DW}	Data Write Time	55	-	ns
t_{DSW}	Write Data Setup Time	15	-	ns
t_{DHW}	Write Data Hold Time	15	-	ns
t_{CLKL}	Clock Low Time	50	-	ns
t_{CLKH}	Clock High Time	50	-	ns
t_R	Rise Time	-	40	ns
t_F	Fall Time	-	40	ns

* (V_{DD} - V_{SS} = 1.65V to 3.5V, T_a = 25°C)





9. Function Specification

(1) UART CONFIGURATION:

ITEM	SETTING VALUE
BAUD RATE	115200
DATA BIT	8
STOP BIT	1
PARITY BIT	NONE
FLOW CONTROL	NONE

(2) COMMAND LIST:

Code	Function	Sequence of HEX command mode through UART	API for Arduino

N/A	Sent a image(192X64 bitmap) to LED Display (An array consist of 1024 bytes bitmap information)	1. A "for" loop to send 1024 bytes user define display information
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			for (i = 0 ; i < 1024; i++) {
			Serial.write(User_define_array[i]);
			}
			while (Serial.read() !='E') {}
			delay(2);

0x80	Write a 5X7 Character	1. Send 0x80
		2. Send which line to put this character
		3. Send which cloumn to put this haracter
		4. Send character's ASCII code
		5. Wait until receive a module available byte ('E') from LED Display
		6. Wait 2ms
			void Write_5X7_Character(int line, int column, int negative, char Char) {
			Serial.write(0x80);
			Serial.write(line);
			Serial.write(column);
			Serial.print(Char);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x81	Write a 8X8 String	1. Send 0x81
		2. Send which line to start the string
		3. Send which cloumn to start the string
		4. Send string
		5. Wait until receive a module available byte('E') from LED Display
		6. Wait 2ms
			void Write_5X7_String(int line, int column, int negative, char * string) {
			Serial.write(0x81);
			Serial.write(line);
			Serial.write(column);
			Serial.print(string);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x82	Write a 8X16 Character	1. Send 0x82
		2. Send which line to put this character
		3. Send which cloumn to put this haracter
		4. Send character's ASCII code
		5. Wait until receive a module available byte('E') from LED Display
		6. Wait 2ms
			void Write_8X16_Character(int line, int column, int negative, char Char) {
			Serial.write(0x82);
			Serial.write(line);
			Serial.write(column);
			Serial.print(Char);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x83	Write a 8X16 String	1. Send 0x83
		2. Send which line to stary the string
		3. Send which cloumn to start the string
		4. Send string
		5. Wait until receive a module available byte('E') from LED Display
		6. Wait 2ms
			void Write_8X16_String(int line, int column, int negative, char * string) {
			Serial.write(0x83);
			Serial.write(line);
			Serial.write(column);
			Serial.print(string);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x84	Display a 8X8 pattern	1. Send 0x84
		2. Send the Up Left X coordinate of pattern
		3. Send the Up Left Y coordinate of pattern
		4. Send the ID of pattern
		5. Wait until receive a module available byte ('E') from LED Display
		6. Wait 2ms
			void Write_8X8_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int negative, int Pattern_ID) {
			Serial.write(0x84);
			Serial.write(Up_Left_Xpos);
			Serial.write(Up_Left_Ypos);
			Serial.write(Pattern_ID);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x85	Display a 8X16 pattern	1. Send 0x85
		2. Send the Up Left X coordinate of pattern
		3. Send the Up Left Y coordinate of pattern
		4. Send the ID of pattern
		5. Wait until receive a module available byte ('E') from LED Display
		6. Wait 2ms
			void Write_8X16_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int negative, int Pattern_ID) {
			Serial.write(0x85);
			Serial.write(Up_Left_Xpos);
			Serial.write(Up_Left_Ypos);
			Serial.write(Pattern_ID);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x86	Display a 16X16 pattern	1. Send 0x86
		2. Send the Up Left X coordinate of pattern
		3. Send the Up Left Y coordinate of pattern
		4. Send the ID of pattern
		5.Wait until receive a module available byte ('E') from LED Display
		6. Wait 2ms
			void Write_16X16_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int negative, int Pattern_ID) {
			Serial.write(0x86);
			Serial.write(Up_Left_Xpos);
			Serial.write(Up_Left_Ypos);
			Serial.write(Pattern_ID);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x87	Display a 32X32 pattern	1. Send 0x87
		2. Send the Up Left X coordinate of pattern
		3. Send the Up Left Y coordinate of pattern
		4. Send the ID of pattern
		5.Wait until receive a module available byte ('E') from LED Display
		6. Wait 2ms
			void Write_32X32_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int negative, int Pattern_ID) {
			Serial.write(0x87);
			Serial.write(Up_Left_Xpos);
			Serial.write(Up_Left_Ypos);
			Serial.write(Pattern_ID);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x90	Draw a line	1. Send 0x90
		2. Send the X coordinate of first point
		3. Send the Y coordinate of first point
		4. Send the X coordinate of second point
		5. Send the Y coordinate of second point
		6. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		7. Wait until receive a module available byte ('E') from LED Display
		8. Wait 2ms
			void Draw_Line(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int negative) {
			Serial.write(0x90);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(X1_Pos);
			Serial.write(Y1_Pos);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x91	Draw a Rectangle	1. Send 0x91
		2. Send the X coordinate of up left corner
		3. Send the Y coordinate of up left corner
		4. Send the X coordinate of bottom right corner
		5. Send the Y coordinate of bottom right corner
		6. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		7. Wait until receive a module available byte ('E') from LED Display
		8. Wait 2ms
			void Draw_Rectangle(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int negative) {
			Serial.write(0x91);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(X1_Pos);
			Serial.write(Y1_Pos);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			}

0x92	Draw a filled Rectangle	1. Send 0x92
		2. Send the X coordinate of up left corner
		3. Send the Y coordinate of up left corner
		4. Send the X coordinate of bottom right corner
		5. Send the Y coordinate of bottom right corner
		6. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		7. Wait until receive a module available byte ('E') from LED Display
		8. Wait 2ms
			void Draw_Filled_Rectangle(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int negative) {
			Serial.write(0x92);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(X1_Pos);
			Serial.write(Y1_Pos);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x93	Draw a Square	1. Send 0x93
		2. Send the X coordinate of up left corner
		3. Send the Y coordinate of up left corner
		4. Send the width of this square
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6.Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw_Square(int X0_Pos, int Y0_Pos, int width, int negative) {
			Serial.write(0x93);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(width);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x94	Draw a Circle	1. Send 0x94
		2. Send the X coordinate of the center
		3. Send the Y coordinate of the center
		4. Send the radius of this circle
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6. Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw_Circle(int X0_Pos, int Y0_Pos, int radius, int negative) {
			Serial.write(0x94);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(radius);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x95	Draw a filled Circle	1. Send 0x95
		2. Send the X coordinate of the center
		3. Send the Y coordinate of the center
		4. Send the radius of this circle
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6. Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw_Filled_Circle(int X0_Pos, int Y0_Pos, int radius, int negative) {
			Serial.write(0x95);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(radius);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x96	Draw a tip upward Triangle	1. Send 0x96
		2. Send the X coordinate of the tip
		3. Send the Y coordinate of the tip
		4. Send the height of the tip to the bottom
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6. Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw_Triangle_Up_Ward(int X0_Pos, int Y0_Pos, int height, int negative) {
			Serial.write(0x96);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(height);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			}

0x97	Draw a filled tip upward Triangle	1. Send 0x97
		2. Send the X coordinate of the tip
		3. Send the Y coordinate of the tip
		4. Send the height of the tip to the bottom
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6. Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw_Filled_Triangle_Up_Ward(int X0_Pos, int Y0_Pos, int height, int negative) {
			Serial.write(0x97);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(height);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x98	Draw a tip downward Triangle	1. Send 0x98
		2. Send the X coordinate of the tip
		3. Send the Y coordinate of the tip
		4. Send the height of the tip to the top
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6. Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw_Triangle_Down_Ward(int X0_Pos, int Y0_Pos, int height, int negative) {
			Serial.write(0x98);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(height);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2); }

0x99	Draw a filled tip downward Triangle	1. Send 0x99
		2. Send the X coordinate of the tip
		3. Send the Y coordinate of the tip
		4. Send the height of the tip to the top
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6.Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw__Filled_Triangle_Down_Ward(int X0_Pos, int Y0_Pos, int height, int negative) {
			Serial.write(0x99);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(height);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x9A	Draw a tip leftward Triangle	1. Send 0x9a
		2. Send the X coordinate of the tip
		3. Send the Y coordinate of the tip
		4. Send the width of the tip to the right
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6.Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw_Triangle_Left_Ward(int X0_Pos, int Y0_Pos, int width, int negative) {
			Serial.write(0x9a);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(width);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x9B	Draw a filled tip leftward Triangle	1. Send 0x9b
		2. Send the X coordinate of the tip
		3. Send the Y coordinate of the tip
		4. Send the width of the tip to the right
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6. Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw_Filled_Triangle_Left_Ward(int X0_Pos, int Y0_Pos, int width, int negative) {
			Serial.write(0x9b);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(width);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x9C	Draw a tip rightward Triangle	1. Send 0x9c
		2. Send the X coordinate of the tip
		3. Send the Y coordinate of the tip
		4. Send the width of the tip to the left
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6. Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw_Triangle_Right_Ward(int X0_Pos, int Y0_Pos, int width, int negative) {
			Serial.write(0x9c);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(width);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x9D	Draw a filled tip rightward Triangle	1. Send 0x9d
		2. Send the X coordinate of the tip
		3. Send the Y coordinate of the tip
		4. Send the width of the tip to the left
		5. Send 1 or 0 for display mode (1 for positive, 0 for negative)
		6.Wait until receive a module available byte ('E') from LED Display
		7. Wait 2ms
			void Draw_Filled_Triangle_Right_Ward(int X0_Pos, int Y0_Pos, int width, int negative) {
			Serial.write(0x9d);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(width);
			Serial.write(0 or 1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x9E	Set a pixel for positive display (show pixel)	1. Send 0x9e
		2. Send the X coordinate of the pixel
		3. Send the Y coordinate of the pixel
		4. Wait until receive a module available byte ('E') from LED Display
		5. Wait 2ms
			void Set_Pixel(int X0_Pos, int Y0_Pos) {
			Serial.write(0x9e);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			while (Serial.read() !='E') {}
			delay(2);
			}

0x9F	Set a pixel for negative display (clear pixel)	1. Send 0x9f
		2. Send the X coordinate of the pixel
		3. Send the Y coordinate of the pixel
		4.Wait until receive a module available byte ('E') from LED Display
		5. Wait 2ms
			void Clear_Pixel(int X0_Pos, int Y0_Pos) {
			Serial.write(0x9f);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xA0	Display image row by row Up Ward	1. Send 0xa0
		2. Send the speed (typical time is 20ms)
		3.Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Display_Row_By_Row_Up_Ward (int Speed) {
			Serial.write(0xa0);
			Serial.write(speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xA1	Display image row by row Down Ward	1. Send 0xa1
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Display_Row_By_Row_Down_Ward (int speed) {
			Serial.write(0xa1);
			Serial.write(speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xA2	Display image column by column Left Ward	1. Send 0xa2
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Display_Column_By_Column_Left_Ward (int speed) {
			Serial.write(0xa2);
			Serial.write(Speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xA3	Display image column by column Right Ward	1. Send 0xa3
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			Void Display_Column_By_Column_Right_Ward (int Speed) {
			Serial.write(0xa3);
			Serial.write(Speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xA4	Erase image row by row Up Ward	1. Send 0xa4
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Erase_Row_By_Row_Up_Ward(int Speed)
			{
			Serial.write(0xa4);
			Serial.write(Speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xA5	Erase image row by row Down Ward	1. Send 0xa5
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Erase_Row_By_Row_Down_Ward(int Speed) {
			Serial.write(0xa5);
			Serial.write(Speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xA6	Erase image column by column Left Ward	1. Send 0xa6
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Erase_Column_By_Column_Left_Ward(int Speed) {
			Serial.write(0xa6);
			Serial.write(Speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xA7	Erase image column by column Right Ward	1. Send 0xa7
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Erase_Column_By_Column_Right_Ward(int Speed) {
			Serial.write(0xa7);
			Serial.write(Speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xA8	Display image Inside Out	1. Send 0xa8
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Display_Inside_Out(int Speed) {
			Serial.write(0xa8);
			Serial.write(Speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xA9	Display image Outside In	1. Send 0xa9
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Display_Outside_In(int Speed) {
			Serial.write(0xa9);
			Serial.write(Speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xAA	Erase image Inside Out	1. Send 0xaa
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Erase_Inside_Out(int Speed) {
			Serial.write(0xaa);
			Serial.write(Speed);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xAB	Erase image Outside In	1. Send 0xab
		2. Send the speed (typical time is 20ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Erase_Outside_In(int Speed) {
			Serial.write(0xab);
			Serial.write(Speed);
			while (Serial.read() !='E') {}
			}

0xD0	Clear display	1. Send 0xd0
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			void Clear_Display_Momery(void) {
			Serial.write(0xd0);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xD1	Show the data in the display memory	1. Send 0xd1
		2.Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			void Show_Display_Momery(void) {
			Serial.write(0xd1);
			while (Serial.read() !='E') {}
			}

0xD2	Scroll the whole display upward	1. Send 0xd2
		2. Send the shift time (typical time is 70ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Scroll_Whole_Display_Memory_Up(int shift time) {
			Serial.write(0xd2);
			Serial.write(shift time);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xD3	Scroll the whole display downward	1. Send 0xd3
		2. Send the shift time (typical time is 70ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Scroll_Whole_Display_Memory_Down(int shift time) {
			Serial.write(0xd3);
			Serial.write(shift time);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xD4	Scroll the whole display leftward	1. Send 0xd4
		2. Send the shift time (typical time is 70ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Scroll_Whole_Display_Memory_Left(int shift time) {
			Serial.write(0xd4);
			Serial.write(shift time);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xD5	Scroll the whole display rightward	1. Send 0xd5
		2. Send the shift time (typical time is 70ms)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Scroll_Whole_Display_Memory_Right(int shift time) {
			Serial.write(0xd5);
			Serial.write(shift time);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xD6	Scroll the section display upward	1. Send 0xd6
		2. Send the X coordinate of up left corner
		3. Send the Y coordinate of up left corner
		4. Send the X coordinate of bottom right corner
		5. Send the Y coordinate of bottom right corner
		6. Send the shift time (typical time is 20ms)
		7. Wait until receive a module available byte ('E') from LED Display
		8. Wait 2ms
			void Scroll_Section_Display_Memory_Up(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int shift time) {
			Serial.write(0xd6);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(X1_Pos);
			Serial.write(Y1_Pos);
			Serial.write(shift time);
			while (Serial.read() !='E') {};
			delay(2);
			}

0xD7	Scroll the section display downward	1. Send 0xd7
		2. Send the X coordinate of up left corner
		3. Send the Y coordinate of up left corner
		4. Send the X coordinate of bottom right corner
		5. Send the Y coordinate of bottom right corner
		6. Send the shift time (typical time is 70ms)
		7. Wait until receive a module available byte ('E') from LED Display
		8. Wait 2ms
			void Scroll_Section_Display_Memory_Down(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int shift time) {
			Serial.write(0xd7);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(X1_Pos);
			Serial.write(Y1_Pos);
			Serial.write(shift time);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xD8	Scroll the section display leftward	1. Send 0xd8
		2. Send the X coordinate of up left corner
		3. Send the Y coordinate of up left corner
		4. Send the X coordinate of bottom right corner
		5. Send the Y coordinate of bottom right corner
		6. Send the shift time (typical time is 20ms)
		7. Wait until receive a module available byte ('E') from LED Display
		8. Wait 2ms
			void Scroll_Section_Display_Memory_Left(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int shift time) {
			Serial.write(0xd8);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(X1_Pos);
			Serial.write(Y1_Pos);
			Serial.write(shift time);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xD9	Scroll the section display rightward	1. Send 0xd9
		2. Send the X coordinate of up left corner
		3. Send the Y coordinate of up left corner
		4. Send the X coordinate of bottom right corner
		5. Send the Y coordinate of bottom right corner
		6. Send the shift time (typical time is 70ms)
		7. Wait until receive a module available byte ('E') from LED Display
		8. Wait 2ms
			void Scroll_Section_Display_Memory_Right(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int shift time) {
			Serial.write(0xd9);
			Serial.write(X0_Pos);
			Serial.write(Y0_Pos);
			Serial.write(X1_Pos);
			Serial.write(Y1_Pos);
			Serial.write(shift time);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xED	Turn Backlight on/off	1. Send 0xed
		2. Send backlight on/off mode
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Turn Backlight on/off {
			Serial.write(0xed);
			Serial.write(mode);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xF0	Turn display Off	1. Send 0xf0
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			void Display_Off(void) {
			Serial.write(0xf0);
			while (Serial.read() !='E') {}
			}

0xF1	Turn display On	1. Send 0xf1
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			void Display_On(void) {
			Serial.write(0xf1);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xF2	Set the brightness of the LED Display	1. Send 0xf2
		2. Send the level of brightness (0~11)
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			void Set_Display_Brightness(int brightness) {
			Serial.write(0xf2);
			Serial.write(brightness);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xF3	Inverse image	1. Send 0xf3
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			void Inverse_Image(void) {
			Serial.write(0xf3);
			while (Serial.read() !='E') {}
			delay(2);
			}

0xF6	Change Instruction mode (0 for Hex Command, 1 for AT Command)	1. Send 0xf6
		2. Send instruction mode 1
		3. Wait until receive a module available byte ('E') from LED Display
		4. Wait 2ms
			int Change_Display_Mode(int mode) {
			Serial.write(0xf6);
			Serial.write(1);
			while (Serial.read() !='E') {}
			delay(2);
			}

ASCII code of 5X7 fonts and 8X16 fonts

Hex	Symbol	Hex	Symbol	Hex	Symbol
0x20		0x40	@	0x60	`
0x21	!	0x41	A	0x61	a
0x22	"	0x42	B	0x62	b
0x23	#	0x43	C	0x63	c
0x24	$	0x44	D	0x64	d
0x25	%	0x45	E	0x65	e
0x26	&	0x46	F	0x66	f
0x27		0x47	G	0x67	g
0x28	(	0x48	H	0x68	h
0x29	)	0x49	I	0x69	i
0x2a	*	0x4a	J	0x6a	j
0x2b	+	0x4b	K	0x6b	k
0x2c	,	0x4c	L	0x6c	l
0x2d	-	0x4d	M	0x6d	m
0x2e	.	0x4e	N	0x6e	n
0x2f		0x4f	O	0x6f	o
0x30	0	0x50	P	0x70	p
0x31	1	0x51	Q	0x71	q
0x32	2	0x52	R	0x72	r
0x33	3	0x53	S	0x73	s
0x34	4	0x54	T	0x74	t
0x35	5	0x55	U	0x75	u
0x36	6	0x56	V	0x76	v
0x37	7	0x57	W	0x77	w
0x38	8	0x58	X	0x78	x
0x39	9	0x59	Y	0x79	y
0x3a	:	0x5a	Z	0x7a	z
0x3b	;	0x5b	]	0x7a	{
0x3c	<	0x5c	\	0x7a	|
0x3d	=	0x5d	[	0x7a	}
0x3e	>	0x5e	^	0x7a	~
0x3f	?	0x5f	_	0x7a	<-

ASCII code of 16X16 fonts

Hex	Symbol
0x30	0
0x31	1
0x32	2
0x33	3
0x34	4
0x35	5
0x36	6
0x37	7
0x38	8
0x39	9

No. of 8X16 pattern

No.	Symbol
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

No. of 8X8 pattern

No.	Symbol
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

No. of 16X16 pattern

No.	Symbol
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

No. of 32X32 pattern

No.	Symbol
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	°C
11	°F
12	

10. Power Reset

4.1 Commands

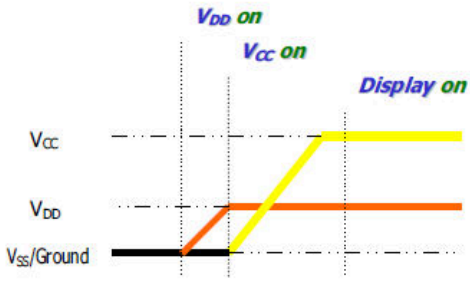
Refer to the Technical Manual for the SSD1309

4.2 Power down and Power up Sequence

To protect OEL panel and extend the panel life time, the driver IC power up/down routine should include a delay period between high voltage and low voltage power sources during turn on/off. It gives the OEL panel enough time to complete the action of charge and discharge before/after the operation.

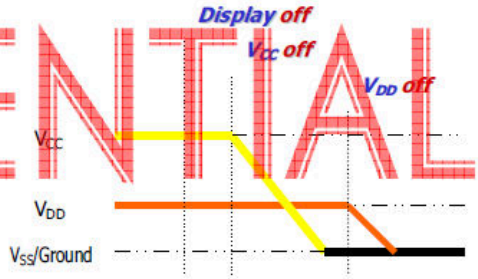
4.2.1 Power up Sequence:

1. Power up V_{DD}
2. Send Display off command
3. Initialization
4. Clear Screen
5. Power up V_{CC}
6. Delay 100ms
   (When V_{CC} is stable)
7. Send Display on command



4.2.2 Power down Sequence:

1. Send Display off command
2. Power down V_{CC}
3. Delay 100ms
   (When V_{CC} is reach 0 and panel is completely discharges)
4. Power down V_{DD}



Note 9:

1) Since an ESD protection circuit is connected between V_{DD} and V_{CC} inside the driver IC, V_{CC} becomes lower than V_{DD} whenever V_{DD} is ON and V_{CC} is OFF.
2) V_{CC} should be kept float (disable) when it is OFF.
3) Power Pins (V_{DD}, V_{CC}) can never be pulled to ground under any circumstance.
4) V_{DD} should not be power down before V_{CC} power down.

4.3 Reset Circuit

When RES# input is low, the chip is initialized with the following status:

1. Display is OFF
2. 128×64 Display Mode
3. Normal segment and display data column and row address mapping (SEG0 mapped to column address 00h and COM0 mapped to row address 00h)
4. Shift register data clear in serial interface
5. Display start line is set at display RAM address 0
6. Column address counter is set at 0
7. Normal scan direction of the COM outputs
8. Contrast control register is set at 7Fh
9. Normal display mode (Equivalent to A4h command)

11. Reliability

The OLED module shall have no failure in the following reliability test. However the following Test of a different item doesn't do by means of the same OLED module.

Test Item	Test Conditions	Note
High temperature operation	70±3℃，t =120hrs	2
Low temperature operation	-40±3℃，t =120hrs	2
High Temperature storage	80±3℃，t =120hrs	1 , 2
Low Temperature storage	-40±3℃，t =120hrs	1 , 2
Temperature Cycle	-30℃ ～ 25℃ ～ 80℃ 30min. 5min. 30min. (1 cycle) Total 10 cycle	1 , 2
Humidity Test	40℃, Humidity 90% , 96 hrs	1 , 2
Vibration Test	Sweep frequency ： 10 ~ 55 , 1.5mmp-p Test direction ： X.Y.Z/3 axis Duration ： 60min/each axis	2
ESD Testing	Contract Discharge Voltage: +1 ~ 5kV and –1 ~ –5kV	3
	Air Discharge Voltage: +1 ~ 8kV and –1 ~ -8kV	

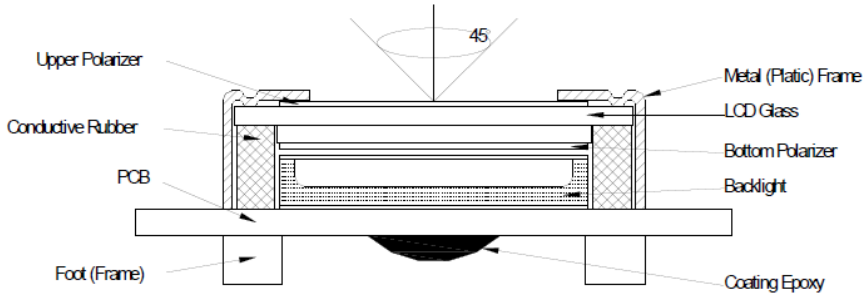
Note l. The module should not have condensation of water on it.
Note 2. The module should be inspected after 4-hour storage under normal Conditions (15℃ to 35℃ , 45% to 65% RH)
Note 3. There will be discharged ten times at every discharging voltage cycle.
Energy Storage Capacitance (Cs+Cd) : 150pF±10%
Discharge Resistance (Rd) : 330Ω±10%
The voltage gap is 1kV.

Definitions of life end point
(1) Current consumption is more than specified value.
(2) Function of the module is not maintained.
(3) There is visible degradation of appearance and display quality.
(4) Contrast ratio is less than 50% of specified minimum value.
(5) Brightness is less than 50% of specified minimum value.

12. Quality

TEST CONDITIONS

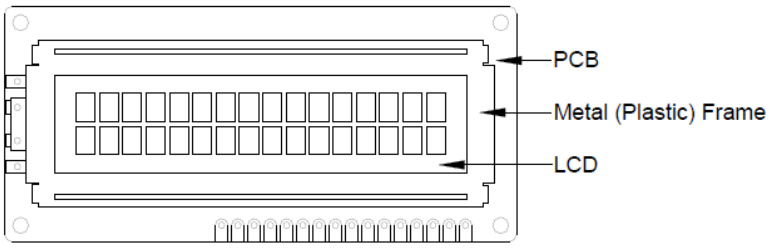
1. Inspection will be done by placing LCM 30cm away from inspector's eyeballs under normal illumination.



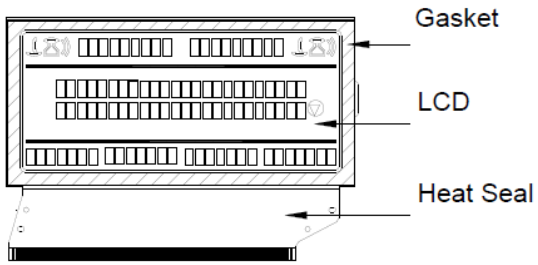
2. View Angle: with in 45°around perpendicular line.

8 - 2. Definition

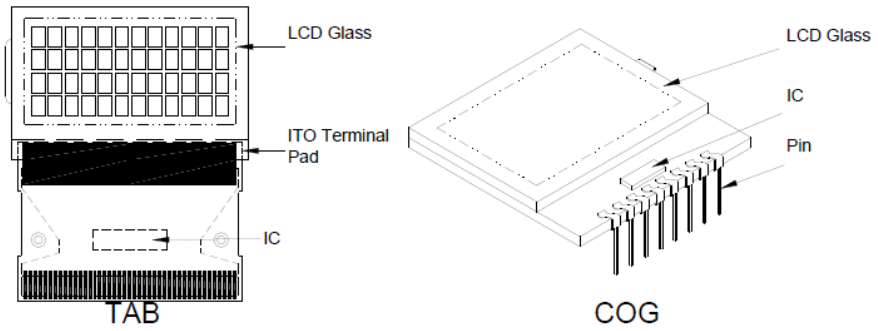
1. COB



2. Heat Seal



3. TAB and COG



QUALITY SPECIFICATIONS (Continued)

8-3. Sampling Plan and Acceptance

1.Sampling Plan

MIL - STD - 105E (||) ordinary single inspection is used.

2.Acceptance

Major defect: AQL = 0.25

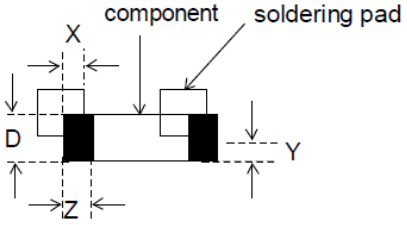
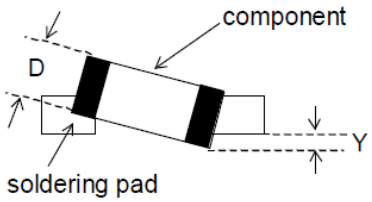
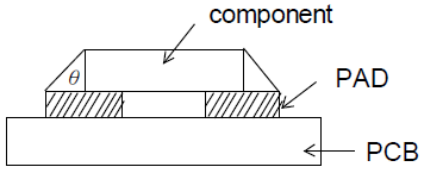
Minor defect: AQL = 0.65

8-4. Criteria

1. COB

Defect	Inspection Item	Inspection Standards	
Major	PCB copper flakes peeling off	Any copper flake in viewing Area should be greater than 1.0mm^2	Reject
Major	Height of coating epoxy	Exceed the dimension of drawing	Reject
Major	Void or hole of coating epoxy	Expose bonding wire or IC	Reject
Major	PCB cutting defect	Exceed the dimension of drawing	Reject

2. SMT

Defect	Inspection Item	Inspection Standards	
Minor	Component marking not readable		Reject
Minor	Component height	Exceed the dimension Of drawing	Reject
Major	Component solder defect (missing, extra, wrong component or wrong orientation		Reject
Minor	Component position shift 	X < 3/4Z Y > 1/3D	Reject Reject
Minor	Component tilt 	Y > 1/3D	Reject
Minor	Insufficient solder 	$\theta \leq 20^{\circ}$	Reject

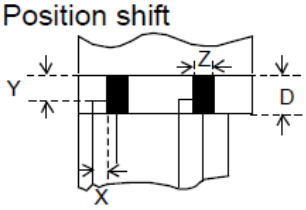
QUALITY SECIFICATIONS (Continued)

8-4. Criteria (Continued)

3. Metal (Plastic) Frame

Defect	Inspection Item	Inspection Standards		
Major	Crack / breakage	Anywhere		Reject
Minor	Frame Scratch	W	L	Acceptable of Scratch
		w<0.03mm	Any	Ignore
		0.03mm≤w<0.25mm	L≤5.0mm	5
		0.25mm≤w<0.4mm	L≤5.0mm	3
		w≥0.4mm	Any	0
		Note: 1. Above criteria applicable to scratch lines with distance greater than 5mm. 2. Scratch on the side face of frame (not visible) can be ignored.		
Minor	Frame Dent, Prick $\Phi = \frac{L+W}{2}$			Acceptable of Dents / Pricks
		Φ≤1.0mm		5
		1.0<Φ≤1.5mm		3
		1.5mm>Φ		0
		Note: 1. Above criteria applicable to any two dents / pricks with distance greater than 5mm 2. Dent / prick on the side face of frame (not visible) can be ignored		
Minor	Frame Deformation	Exceed the dimension of drawing		
Minor	Metal Frame Oxidation	Any rust		

4. Flexible Film Connector (FFC)

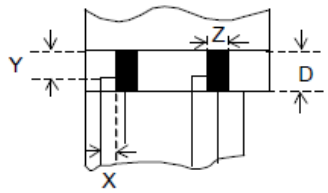
Defect	Inspection Item		Inspection Standards	
Minor	Tilted soldering		Within the angle ±3°	Acceptable
Minor	Uneven solder joint /bump			Reject
Minor	Hole	$\Phi = \frac{L+W}{2}$	Expose the conductive line	Reject
			Φ > 1.0mm	Reject
Minor	Position shift  		Y > 1/3D	Reject
			X > 1/2Z	Reject

8-4. Criteria (Continued)

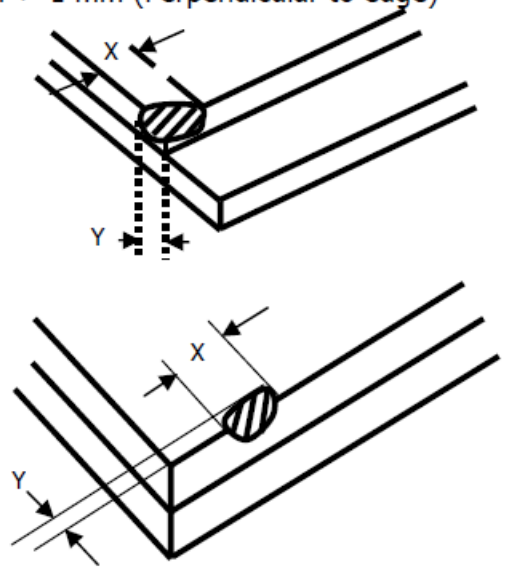
5. Screw

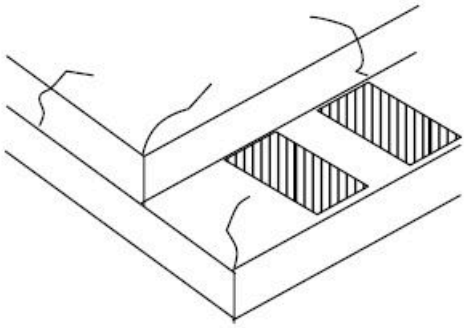
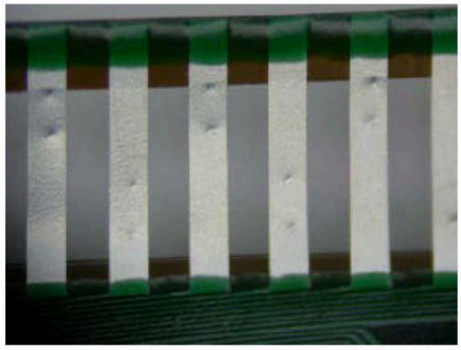
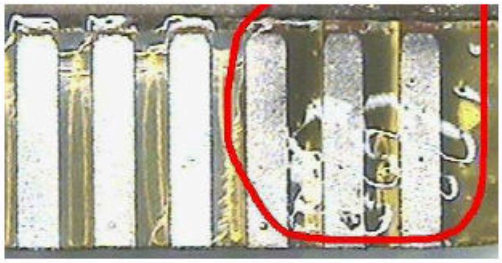
Defect	Inspection Item	Inspection Standards	
Major	Screw missing/loosen		Reject
Minor	Screw oxidation	Any rust	Reject
Minor	Screw deformation	Difficult to accept screw driver	Reject

6. Heat seal 、TCP 、FPC

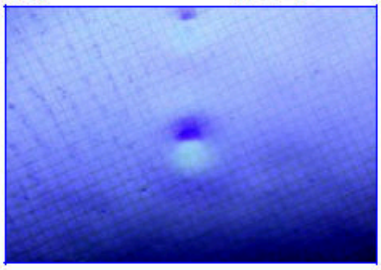
Defect	Inspection Item	Inspection Standards	
Major	Scratch expose conductive layer		Reject
Minor	HS Hole $\Phi = \frac{L + W}{2}$	Φ > 0.2mm	Reject
Major	Adhesion strength	Less than the specification	Reject
Minor	Position shift 	Y > 1/3D	Reject
		X > 1/2Z	Reject
Major	Conductive line break		Reject

7. Inspection Specification of OLED modules

Check Item	Classification	Criteria
Panel General Chipping	Minor	X > 6 mm (Along with Edge) Y > 1 mm (Perpendicular to edge) 

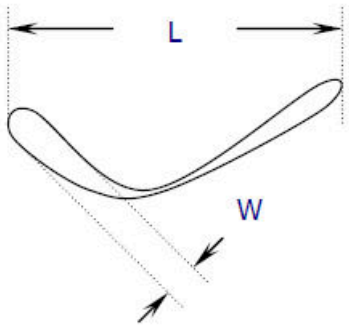
Check Item	Classification	Criteria
Panel Crack	Minor	Any crack is not allowable. 
Copper Exposed (Even Pin or Film)	Minor	Not Allowable by Naked Eye Inspection
Film or Trace Damage	Minor	
Terminal Lead Prober Mark	Acceptable	
Glue or Contamination on Pin (Couldn't Be Removed by Alcohol)	Minor	
Ink Marking on Back Side of panel (Exclude on Film)	Acceptable	Ignore for Any

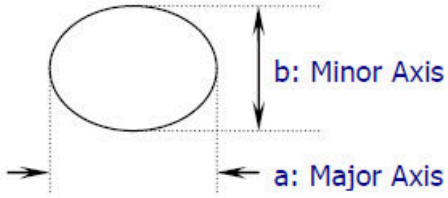
It is recommended to execute in clear room environment (class 10k) if actual in necessary.

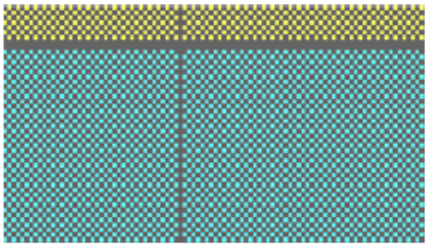
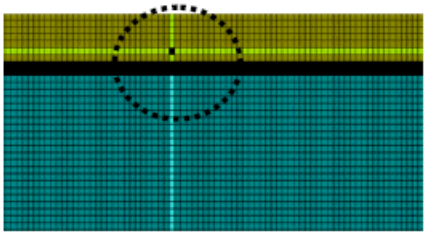
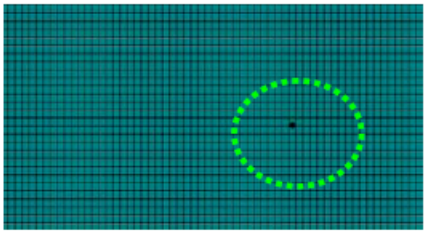
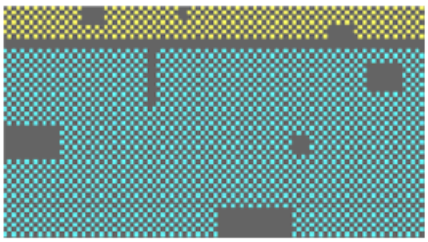
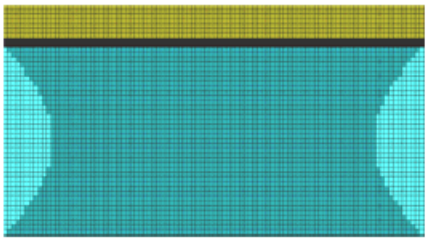
Check Item	Classification	Criteria
Any Dirt & Scratch on Polarizer's Protective Film	Acceptable	Ignore for not Affect the Polarizer
Scratches, Fiber, Line-Shape Defect (On Polarizer)	Minor	W ≤ 0.1 Ignore W > 0.1 L ≤ 2 n ≤ 1 L > 2 n = 0
Dirt, Black Spot, Foreign Material, (On Polarizer)	Minor	Φ ≤ 0.1 Ignore 0.1 < Φ ≤ 0.25 n ≤ 1 0.25 < Φ n = 0
Dent, Bubbles, White spot (Any Transparent Spot on Polarizer)	Minor	Φ ≤ 0.5 ➔ Ignore if no Influence on Display 0.5 < Φ n = 0 
Fingerprint, Flow Mark (On Polarizer)	Minor	Not Allowable

* Protective film should not be tear off when cosmetic check.

** Definition of W & L & Φ (Unit: mm): Φ = (a + b) / 2





Check Item	Classification	Criteria
No Display	Major	
Missing Line	Major	
Pixel Short	Major	
Darker Pixel	Major	
Wrong Display	Major	
Un-uniform	Major	

13. Handling Precaution

1. Mounting Method

-The panel of the OLED Module consists of two thin glass plates with polarizes which easily get damaged since the Module is fixed by utilizing fitting holes in the printed circuit board. Extreme care should be taken when handling the OLED Modules.

2. Caution of OLED handling & cleaning

-When cleaning the display surface, use soft cloth with solvent (recommended below) and Wipe lightly.

*Isopropyl alcohol

*Ethyl alcohol

*Trichlorotriflorothane

-Do not wipe the display surface with dry or hard materials that will damage the polarize surface. Do not use the following solvent：

*Water

* Acetone

*Aromatics

3. Caution against static charge

-The OLED Module use C-MOSLSI drivers, so we recommend end that you connect any unused input terminal to VDD or VSS, do not input any signals before power is turned on. And ground your body, Work/assembly table. And assembly equipment to protect against static electricity.

4. Packaging

-Modules use OLED elements, and must be treated as such. Avoid in tense shock and falls from a height.

-To prevent modules from degradation. Do not operate or store them exposed directly to sunshine or high temperature/humidity.

5. Caution for operation

-It is indispensable to drive OLED's with in the specified voltage limit since the higher voltage than the limit shorten OLED life.

-An electrochemical reaction due to direct current causes OLED deterioration, Avoid the use of direct current drive.

-Response time will be extremely delayed at lower temperature than the operating temperature range and on the other hand at higher temperature OLED's show dark color in them. However those phenomena do not mean malfunction or out of order with OLED's. Which will come back in the specified operating temperature range.

-If the display area is pushed hard during operation, some font will be abnormally displayed but it resumes normal condition after turning off once.

A slight dew depositing on terminals is a cause for electro-chemical reaction resulting in terminal open circuit. Usage under the relative condition of 40 ℃, 50%RH or less is required.

(1)Soldering condition to I/O PIN terminals

Temperature at tip of the iron: 280±10℃

Soldering time: 3~4 sec.

Type of solder: Eutectic solder (containing colophony-flux)

*Please do not use flux because it may soak into OLED Module or contaminate it.

6. Storage

-In the case of storing for a long period of time (for instance. For years) for the purpose or replacement use, The following ways are recommended.

-Storage in a polyethylene bag with sea OLED so as not to enter fresh air outside in it, And with no desiccant.

-Placing in a dark place where neither exposure to direct sunlight nor light is. Keeping temperature in the specified storage temperature range.

-Storing with no touch on polarizer surface by the anything else. (It is recommended to store them as they have been contained in the inner container at the time of delivery)

7. Safety

-It is recommendable to crash damaged or unnecessary OLED into pieces and wash off liquid crystal by using solvents such as acetone and ethanol. Which should be burned up later.

-When any liquid crystal leaked out of a damaged glass cell comes in contact with your hands, please wash it off well with soap and water.

14. Packing Method

T.B.D