AZ DISPLAYS, INC

SPECIFICATIONS FOR
LIQUID CRYSTAL DISPLAY

CUSTOMER APPROVAL			
※ PART NO. : ACM2002EI-FL-YBW VER R1			
APPROVAL		COMPANY CHOP	
CUSTOMER COMMENTS			

DISPLAYTRONIC ENGINEERING APPROVAL		
DESIGN BY	CHECKED BY	APPROVED BY

REVISION RECORD

REVISION	REVISION DATE	PAGE	CONTENTS
VER1	**28 /12-2009**		**First Issue**

※ CONTENTS

1.0 GENERAL SPECS
2.0 ABSOLUTE MAXIMUM RATINGS
3.0 ELECTRICAL CHARACTERISTICS
4.0 OPTICAL CHARACTERISTICS
5.0 BLOCK DIAGRAM
6.0 PIN ASSIGNMENT
7.0 I2C TIMING CHARACTERISTICS
8.0 REFERENCE CODE
9.0 MECHANICAL DIAGRAM
10.0 RELIABILITY TEST
11.0 DISPLAY INSTRUCTION TABLE
12.0 STANDARD CHARACTER PATTERNS
13.0 PRECAUTION FOR USING LCM

1.0 GENERAL SPECS

1. Display Format	20*2 Character
2. Power Supply	5.0V
3. Overall Module Size	116.0mm(W) x 37.0mm(H) x max 13.5mm(D)
4. Viewing Aera(W*H)	83.0mm(W) x 18.6mm(H)
5. Dot Size (W*H)	0.60mm(W) x0.65mm(H)
6. Dot Pitch (W*H)	0.65mm(W) x 0.70mm(H)
7. Character Size (W*H)	3.20mm(W) x 5.55mm(H)
8. Character Pitch (W*H)	3.70mm(W) x 5.95mm(H)
9. Viewing Direction	6:00 O'Clock
10. Driving Method	1/16Duty,1/5Bias
11. Controller IC	SPLC780D1-00A-C
12. LC Fluid Options	STN（Y-G）/POSITIVE
13. Polarizer Options	TRANSFLECTIVE
14. Backlight Options	Y-G
15. Operating temperature	-20ºC ~ 70ºC
16. Storage temperature	-30ºC ~ 80ºC
17. ROHS	ROHS compliant

2.0 ABSOLUTE MAXIMUM RATINGS

Item Sy	mbol	Min	Typ	Max	Unit
Operating temperature	Top	-20	--	70	ºC
Storage temperature	Tst	-30	--	80	ºC
Input voltage	Vin	Vss-0.3	--	Vdd+0.3	V
Supply voltage for logic	Vdd- Vss	2.7	--	5.5	V
Supply voltage for LCD drive	Vdd- Vo	3.0	--	8.0	V

3.0 ELECTRICAL CHARACTERISTICS

3.0.1 Electrical Characteristics Of LCM

Item Sy	mbol	Condition	Min	Typ	Max	Unit
Power Supply Voltage	Vdd	25ºC	--	5.0	--	V
Power Supply Current	Idd	Vdd=5.0V, Fi2c=100kHz (not including BackLight)	--	3.5	5	mA
Input voltage (high)	Vih	Pins:(SCL,SDA) VDD=5V	2.5	--	Vdd	V
Input voltage (low)	Vil		-0.3	--	0.6	V
Recommended LC Driving Voltage	Vdd -Vo	25ºC	4.3	4.5	4.7	V
		70ºC	3.9	4.1	4.3	

3.0.2 The Characteristics Of LED Backlight(yellow-green)

3.0.2.1 Electrical-Optical Characteristics Of LED Backlight (Ta=25ºC)

Item Sy	mbol	Condition	Min	Typ	Max	Unit
Forward Current	If	VF=4.1±0.2V	--	140	--	mA
Reverse Voltage	Vr		--	--	10	V
Luminance Lv		If=140mA	175	190	--	cd/
Uniformity	Δ	(Lvmin/Lvmax)%	75%	90%	--	
Spectral line half width	Δλ	If=1 40mA	--	30	--	nm
Lifetime		If=140mA	18000	20000	---	Hours

NOTE:

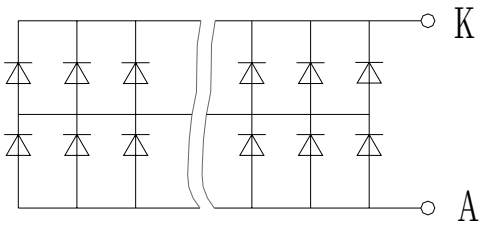
(1)The luminance is the average value of 5 points,The measurement instrument is BM-7 luminance colorimeter.The diameter of aperture is Φ5mm

(2) Luminance means the backlight brightness without glass.

(3) Backlight lifetime means luminance value larger than half the original after 20000 hours' continuous work.

3.0.2.2 Backlight Circuit

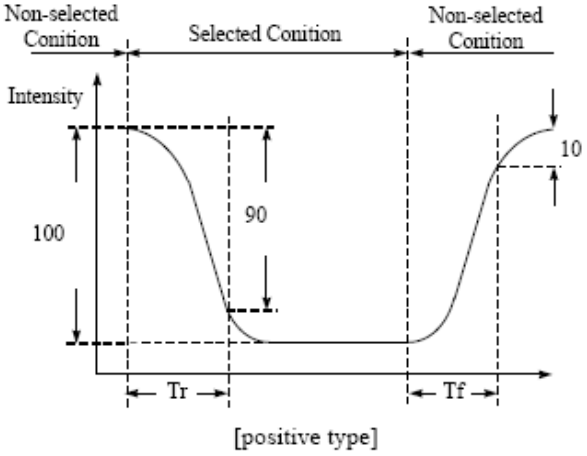
电路图（LED 2X14=28 Dies）

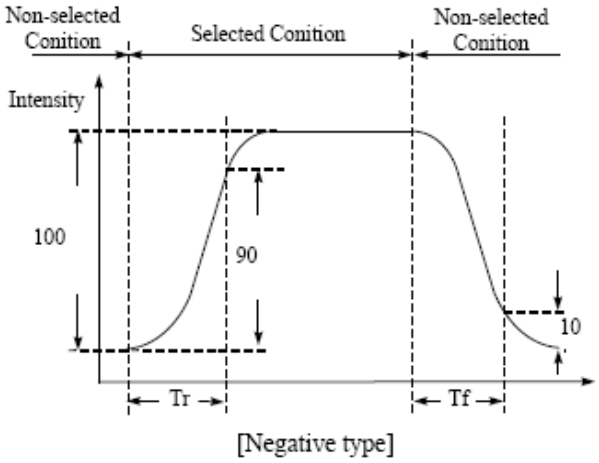


4.0 OPTICAL CHARACTERISTICS (Ta=25ºC, Vdd= 5.0V±0.25V)

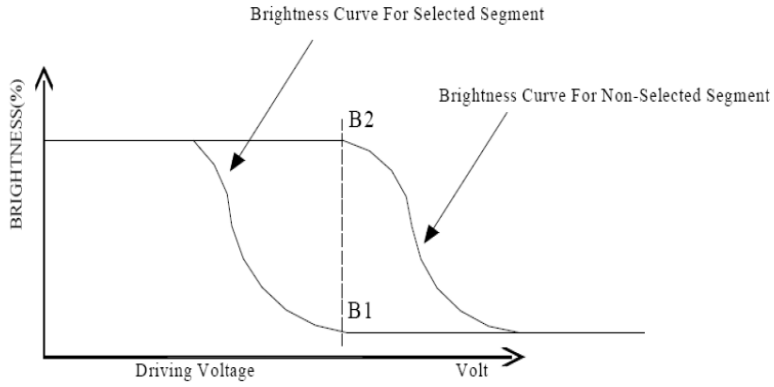
Item Sy	mbol	Condition	Min	Typ	Max	Unit
Viewing angle (horizontal)		Cr 2.0	-35	-	35	deg
Viewing angle (vertical)		Cr 2.0	-25	-	40	deg
Contrast Ratio	Cr	=0 , =0	-	6	-	
Response time (rise)	Tr	=0 , =0	-	18 0	300	ms
Response time (fall)	Tf	=0 , =0	-	15 0	250	ms

(1). Definition of Optical Response Time



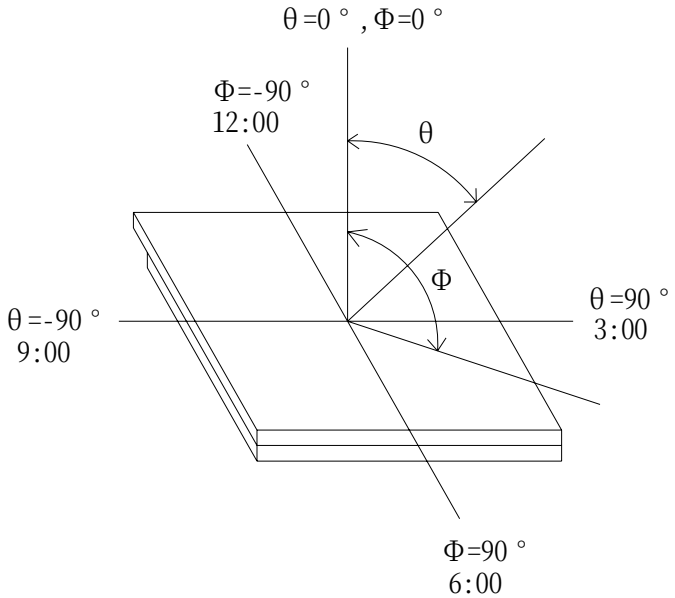


(2). Definition of Contrast Ratio

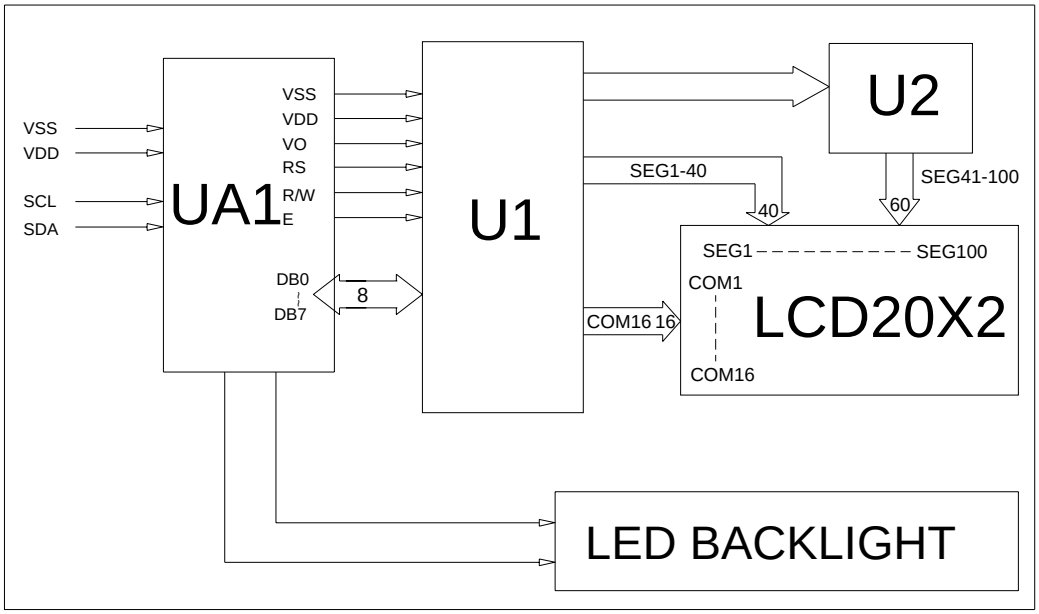


$$Cr = \frac{\text{Brightness of Non-selected Segment(B2)}}{\text{Brightness of selected Segment(B1)}}$$

(3). Definition of Viewing Angle and



5.0 BLOCK DIAGRAM



6.0 PIN ASSIGNMENT

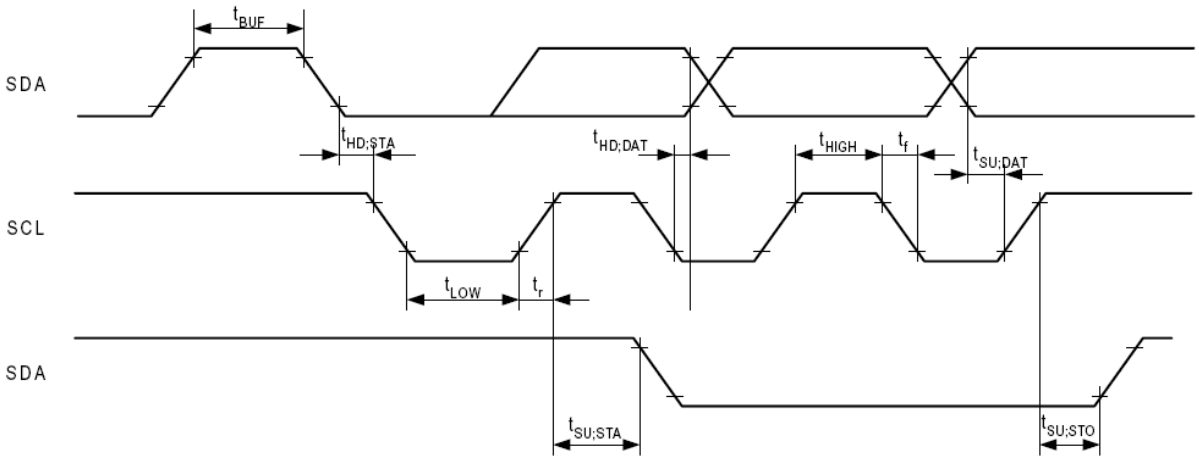
Pin No.	Symbol	Function
1	VSS	Ground
2	V DD	+5.0V
3	SCL	SERIAL CLOCK INPUT
4	SDA	SERIAL DATA INPUT
5-16	TEST	FOR TEST,DON'T USE

7.0 I2C TIMING CHARACTERISTICS

7.0.1 TIMING

V_{DD} = 5V , V_{SS} = 0V, T_A = 25°C, unless otherwise specified

Symbol	Parameter	Min.	Typ.	Max.	Unit	Conditions
Timing characteristics: I²C-bus interface (input capacitance C_I = 10pF)						
f_{SCL}	SCL clock frequency	–	–	100	KHz	
t_{LOW}	SCL clock low period	5.2	–	–	µs	
t_{HIGH}	SCL clock high period	2.4	–	–	µs	
$t_{SU;DAT}$	data set-up time	400	–	–	ns	
$t_{HD;DAT}$	data hold time	0	–	–	ns	
t_r	SCL, SDA rise time	–	–	400	ns	
t_f	SCL, SDA fall time	–	–	400	ns	
C_B	capacitive bus line load	–	–	400	pF	
$t_{SU;STA}$	set-up time for a repeated START condition	2.4	–	–	µs	
$t_{HD;STA}$	START condition hold time	2.4	–	–	µs	
$t_{SU;STO}$	set-up time for STOP condition	2.4	–	–	µs	
t_{SW}	tolerable spike width on bus	–	–	80	ns	



7.0.2 I2C interface:

It just only could write Data or Instruction to LCM by the IIC Interface.
It could not read Data or Instruction from LCM (except Acknowledge signal).
SCL: serial clock input
SDA: serial data input
Slaver address could only set to 1010000, no other slaver address could be set.

The I2C interface send RAM data and executes the commands sent via the I2C Interface. It could send data bit to the RAM. The I2C Interface is two-line communication between different ICs or modules. The two lines are a Serial Data line (SDA) and a Serial Clock line (SCL). Data transfer may be initiated only when the bus is not busy.

BIT TRANSFER:

One data bit is transferred during each clock pulse. The data on the SDA line must remain stable during the HIGH period of the clock pulse because changes in the data line at this time will be interpreted as a control signal. Bit transfer is illustrated in Fig.1.

START AND STOP CONDITIONS:

Both data and clock lines remain HIGH when the bus is not busy. A HIGH-to-LOW transition of the data line, while the clock is HIGH is defined as the START condition (S). A LOW-to-HIGH transition of the data line while the clock is HIGH is defined as the STOP condition (P). The START and STOP conditions are illustrated in Fig.2.

SYSTEM CONFIGURATION:

The system configuration is illustrated in Fig.3.

Transmitter: the device, which sends the data to the bus

Master: the device, which initiates a transfer, generates clock signals and terminates a transfer

Slave: the device addressed by a master

Multi-Master: more than one master can attempt to control the bus at the same time without corrupting the message

Arbitration: procedure to ensure that, if more than one master simultaneously tries to control the bus, only one is allowed to do so and the message is not corrupted

Synchronization: procedure to synchronize the clock signals of two or more devices.

ACKNOWLEDGE:

Acknowledge is not Busy Flag in I2C interface.

Each byte of eight bits is followed by an acknowledge bit. The acknowledge bit is a HIGH signal put on the bus by the transmitter during which time the master generates an extra acknowledge related clock pulse. A slave receiver which is addressed must generate an acknowledge after the reception of each byte. A master receiver must also generate an acknowledge after the reception of each byte that has been clocked out of the slave transmitter. The device that acknowledges must pull-down the SDA line during the acknowledge clock pulse, so that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse (set-up and hold times must be taken into consideration). A master receiver must signal an end-of-data to the transmitter by not generating an acknowledge on the last byte that has been clocked out of the slave. In this event the transmitter must leave the data line HIGH to enable the master to generate a STOP condition. Acknowledgement on the I2C Interface is illustrated in Fig.4.

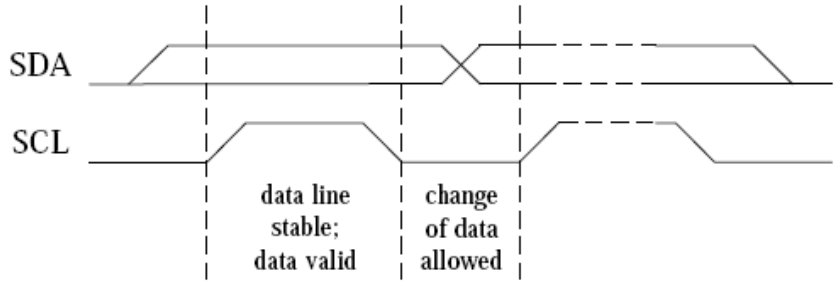


Figure 1. Bit transfer

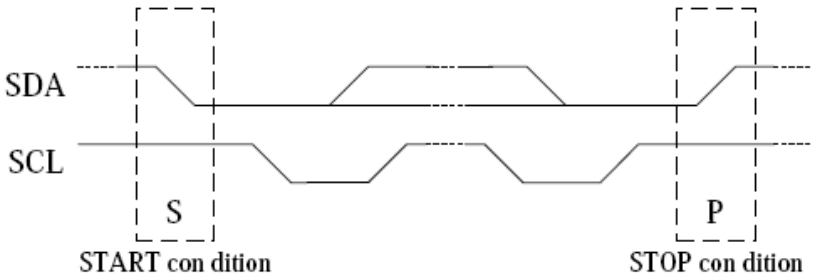


Figure 2. Definition of START and STOP conditions

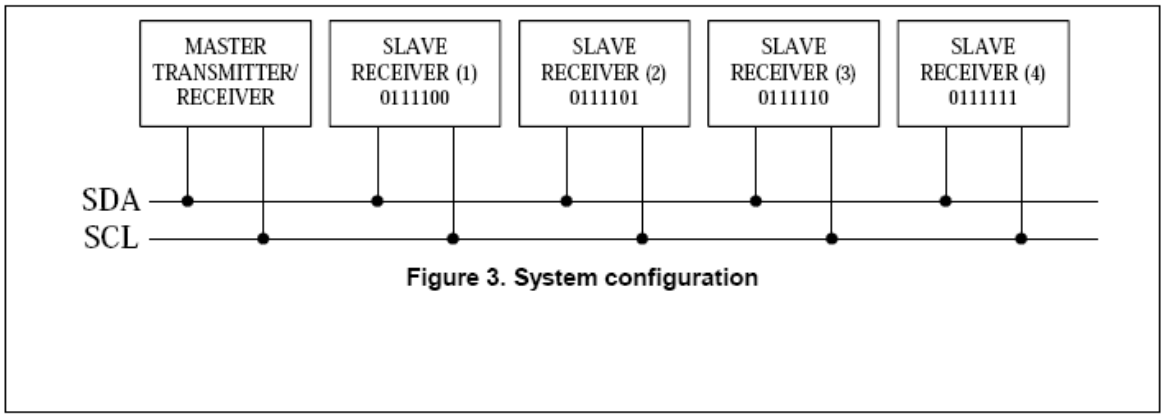


Figure 3. System configuration

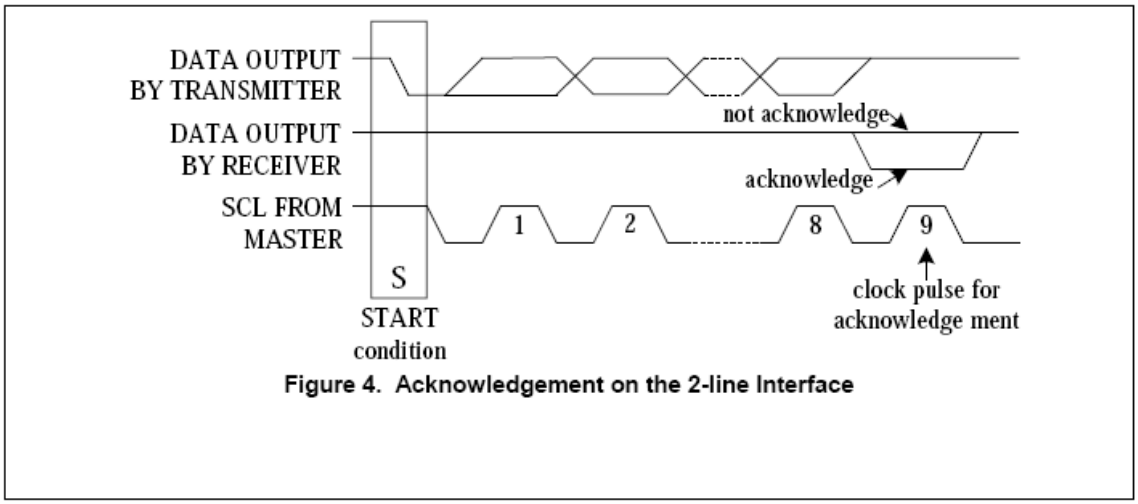


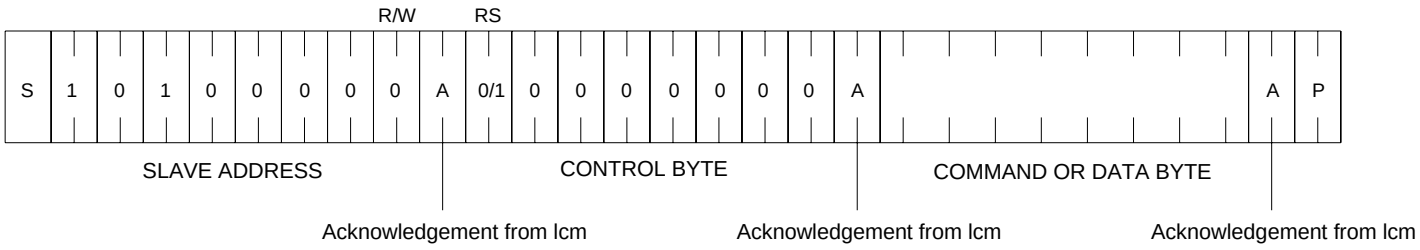
Figure 4. Acknowledgement on the 2-line Interface

7.0.3 I2C Interface protocol:

The LCM supports command, data write addressed slaves on the bus.

Before any data is transmitted on the I2C Interface, the device, which should respond, is addressed first. Only one 7-bit slave addresses (1010000) is reserved for the LCM. The R/W is assigned to 0 for Write only. The I2C Interface protocol is illustrated in Fig.5.

The sequence is initiated with a START condition (S) from the I2C Interface master, which is followed by the slave address.All slaves with the corresponding address acknowledge in parallel, all the others will ignore the I2C Interface transfer. After acknowledgement, a control byte will be followed which defines RS bit. The state of the RS bit defines whether the data byte is interpreted as a command or as RAM data. When the control byte is 0x00,the next folling byte is command byte.When the control byte is 0x80 ,the next folling byte is data byte. Only the addressed slave makes the acknowledgement after each byte. At the end of the transmission the I2C INTERFACE-bus master issues a STOP condition (P).



8.0 THE REFERENCED CODE

8.0.1 PIC C REFERENCE CODE FOR PIC SERIES MCU

```
//PIC16F877A,11.0592MHZ,PICC
//*****************************************************************************//
#include <pic.h>
#include <pic1687x.h>
//*****************************************************************************//
#define uchar unsigned char
//*****************************************************************************//
#define sda  RC4
#define sclk RC3
//*****************************************************************************//
__CONFIG(HS & WDTDIS & PWRTDIS & BORDIS & LVPDIS & DUNPROT & UNPROTECT);
//*****************************************************************************//
const uchar table[]=
{
  "This is program for "
  "The Master is:16f877"
  "i2c interface.      "
  "The Slave is:16f689 "
};
//*****************************************************************************//
void long_delay(void)
{
  uchar i,j;
  for(i=0;i<25;i++)
    for(j=0;j<255;j++)
      ;
}
//*****************************************************************************//
void short_delay(void)
{
  uchar i,j;
  for(i=0;i<1;i++)
    for(j=0;j<1;j++)
      ;
}
//*****************************************************************************//
void start(void)
{
  sclk=1;
  sda=1;
  sda=0;
  sclk=0;
}
//*****************************************************************************//
void stop(void)
{
  sclk=1;
  sda=0;
  sda=1;
  sclk=0;
}
//*****************************************************************************//
void write_byte(byte)
{
  uchar i;
  for(i=8;i>0;i--)
```

```
   {
   if((byte & 0x80)==0x80)
     sda=1;
   else
     sda=0;
   sclk=1;
   sclk=0;
   byte=byte<<1;
   }
  sda=1;
  sclk=1;
  TRISC4=1;
  while(sda);
  sclk=0;
  TRISC4=0;
  short_delay();                    //wait for a moment,you can adjust it.
}
//*************************************************************************************//
void WtiteCommand(command)
{
  start();
  write_byte(0xa0);
  write_byte(0x00);
  write_byte(command);
  stop();
}
//*************************************************************************************//
void WriteData(data)
{
  start();
  write_byte(0xa0);
  write_byte(0x80);
  write_byte(data);
  stop();
}
//*************************************************************************************//
main()
{
  uchar i;
  long_delay();                     //at least 15ms
  long_delay();
  POR=1;
  BOR=1;
  TRISC=0x00;
  sda=1;
  sclk=1;
  WtiteCommand(0x01);
  long_delay();                     //at least 5ms
  long_delay();
  WtiteCommand(0x38);
  long_delay();                     //at least 5ms
  long_delay();
  WtiteCommand(0x0f);
  long_delay();                     //at least 5ms
  long_delay();
  WtiteCommand(0x06);
  long_delay();                     //at least 5ms
  long_delay();
  while(1)
  {
  WtiteCommand(0x80);
```

```
long_delay();                    //at least 5ms
long_delay();
  for(i=0;i<80;i++)
    {
    WriteData(table[i]);
    //long_delay();
    }
  WtiteCommand(0x01);
  }
}
//*************************************************************************************//
```

8.0.2 KEIL C REFERENCE CODE FOR 51 SERIES MCU

```
//AT89S52,12MHZ,KEIL
//*************************************************************************************//
#include"at89x51.h"
//*************************************************************************************//
#define sclk  P2_0
#define sda   P2_1
//*************************************************************************************//
#define uchar unsigned char
//*************************************************************************************//
uchar code table[]=
{
  "This is program for "
  "The Master is:89S52 "
  "i2c interface.      "
  "The Slave is:16f689 "
};
//*************************************************************************************//
void long_delay(void)
{
  uchar i,j;
  for(i=0;i<25;i++)
    for(j=0;j<255;j++)
      ;
}
//*************************************************************************************//
void short_delay(void)
{
  uchar i,j;
  for(i=0;i<1;i++)
    for(j=0;j<1;j++)
      ;
}
//*************************************************************************************//
void start(void)
{
  sclk=1;
  sda=1;
  sda=0;
  sclk=0;
}
//*************************************************************************************//
void stop(void)
{
  sclk=1;
  sda=0;
  sda=1;
  sclk=0;
}
//*************************************************************************************//
void write_byte(byte)
{
  uchar i;
  for(i=8;i>0;i--)
    {
    if((byte & 0x80)==0x80)
      sda=1;
    else
      sda=0;
    sclk=1;
```

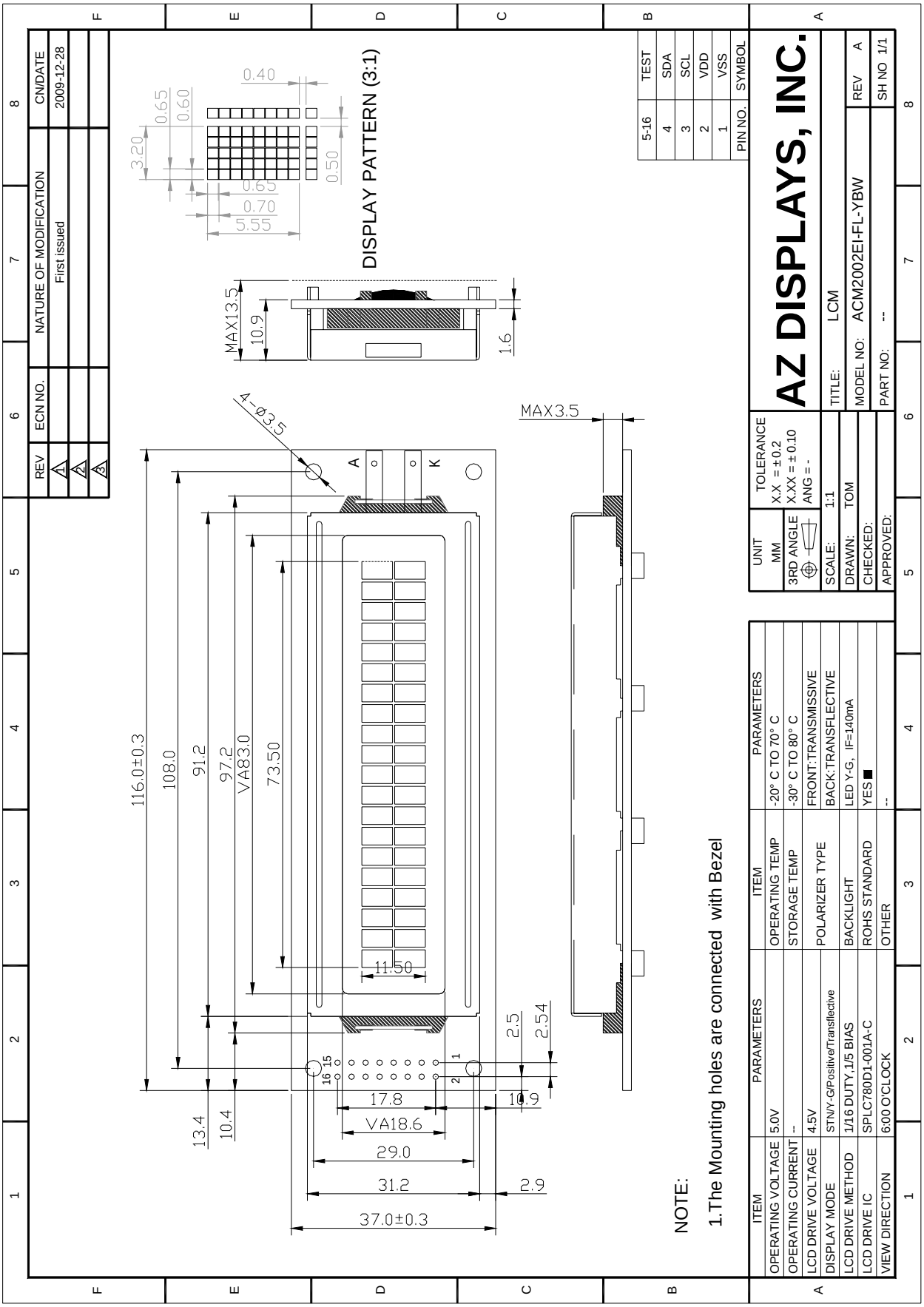
```
   sclk=0;
   byte=byte<<1;
   }
 sda=1;
 sclk=1;
 while(sda);
 sclk=0;
 short_delay();                    //wait for a moment ,you can adjust it.
}
//*************************************************************************************//
void WtiteCommand(command)
{
 start();
 write_byte(0xa0);
 write_byte(0x00);
 write_byte(command);
 stop();
}
//*************************************************************************************//
void WriteData(dat)
{
 start();
 write_byte(0xa0);
 write_byte(0x80);
 write_byte(dat);
 stop();
}
//*************************************************************************************//
main()
{
 uchar i;
 long_delay();                     //at least 15ms
 long_delay();
 sda=1;
 sclk=1;
 WtiteCommand(0x01);
 long_delay();                     //at least 5ms
 long_delay();
 WtiteCommand(0x38);
 long_delay();                     //at least 5ms
 long_delay();
 WtiteCommand(0x0f);
 long_delay();                     //at least 5ms
 long_delay();
 WtiteCommand(0x06);
 long_delay();                     //at least 5ms
 long_delay();
 while(1)
 {
 WtiteCommand(0x80);
 long_delay();                     //at least 5ms
 long_delay();
 for(i=0;i<80;i++)
   {
   WriteData(table[i]);
   long_delay();
   }
 WtiteCommand(0x01);
 }
}
//*************************************************************************************//
```

8.0.3 PIC C REFERENCE CODE FOR PIC SERIES MCU(USING MSSP MODULE)

```
//PIC16F877A,11.0592MHZ,PICC
//*****************************************************************************//
#include <pic.h>
#include <pic1687x.h>
//*****************************************************************************//
#define uchar unsigned char
//*****************************************************************************//
#define sda  RC4
#define sclk RC3
//*****************************************************************************//
__CONFIG(HS & WDTDIS & PWRTDIS & BORDIS & LVPDIS & DUNPROT & UNPROTECT);
//*****************************************************************************//
const uchar table[]=
{
  "This is program for "
  "The Master is:16f877"
  "i2c interface.      "
  "The Slave is:16f689 "
};
//*****************************************************************************//
void long_delay(void)
{
  uchar i,j;
  for(i=0;i<25;i++)
    for(j=0;j<255;j++)
      ;
}
//*****************************************************************************//
void short_delay(void)
{
  uchar i,j;
  for(i=0;i<1;i++)
    for(j=0;j<1;j++)
      ;
}
//*****************************************************************************//
void start(void)
{
  SEN=1;
  while(!SSPIF);
  SSPIF=0;
}
//*****************************************************************************//
void stop(void)
{
  PEN=1;
  while(!SSPIF);
  SSPIF=0;
}
//*****************************************************************************//
void write_byte(byte)
{
  SSPBUF=byte;
  while(!SSPIF);
  while(ACKSTAT);
  SSPIF=0;
}
//*****************************************************************************//
```

```
void WtiteCommand(command)
{
 start();
 write_byte(0xa0);
 write_byte(0x00);
 write_byte(command);
 stop();
}
//*************************************************************************************//
void WriteData(data)
{
 start();
 write_byte(0xa0);
 write_byte(0x80);
 write_byte(data);
 stop();
}
//*************************************************************************************//
main()
{
 uchar i;
 long_delay();                  //delay at least 15ms
 long_delay();
 POR=1;
 BOR=1;
 SSPSTAT=0x00;
 TRISC=0x18;
 SSPADD=0x09;
 SSPIE=0;
 SSPCON=0x28;
 INTCON=0x00;
 WtiteCommand(0x01);
 long_delay();                   //delay at least 5ms
 WtiteCommand(0x38);
 long_delay();                   //delay at least 5ms
 WtiteCommand(0x0f);
 long_delay();                   //delay at least 5ms
 WtiteCommand(0x06);
 long_delay();                   //delay at least 5ms
 while(1)
 {
 WtiteCommand(0x80);
 long_delay();                    //delay at least 5ms
 for(i=0;i<80;i++)
   {
   WriteData(table[i]);
   long_delay();                 //delay at least 5ms
   }
 WtiteCommand(0x01);
 }
}
//*************************************************************************************//
```

9.0 MECHANICAL DIAGRAM



10.0 RELIABILITY TEST

NO	Test Item		Description	Test Condition	Remark
1	Environmental Test	High temperature storage	Applying the high storage temperature Under normal humidity for a long time Check normal performance	80 ºC 96hrs	
2		Low temperature storage	Applying the low storage temperature Under normal humidity for a long time Check normal performance	-30ºC 96hrs	
3		High temperature Operation	Apply the electric stress(Volatge and current) Under high temperature for a long time	70 ºC 96hrs	Note1
4		Low temperature Operation	Apply the electric stress Under low temperature for a long time	-20ºC 96hrs	Note1 Note2
5		High temperature/High Humidity Storage	Apply high temperature and high humidity storage for a long time	90% RH 40ºC 96hrs	Note2
6		Temperature Cycle	Apply the low and high temperature cycle -30ºC＜＞25ºC＜＞80ºC ＜＞25ºC 30min 10min 30min 10min ←——— 1 cycle ———→ Check normal performance	-30ºC/80ºC 10 cycle	
7	Mechanical Test	Vibration test(Package state)	Applying vibration to product check normal performance	Freq:10-55Hz Max Acceleration 5G 1cycle time:1min time X.Y.Z direction for 15 mines	
8		Shock test(package state)	Applying shock to product check normal performance	Drop them through 70cm height to strike horizontal plane	
9	Other				

Remark

Note1:Normal operations condition (25ºC±5ºC).

Note2:Pay attention to keep dewdrops from the module during this test.

11.0 DISPLAY INSTRUCTION TABLE

Instruction	Instruction Code										Description	Execution time (Temp = 25℃)		
	RS	RW	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		Fosc= 190KHz	Fosc= 270KHz	Fosc= 350KHz
Clear Display	0	0	0	0	0	0	0	0	0	1	Write "20H" to DDRAM and set DDRAM address to "00H" from AC	2.16ms	1.52ms	1.18ms
Return Home	0	0	0	0	0	0	0	0	1	-	Set DDRAM address to "00H" from AC and return cursor to its original position if shifted. The contents of DDRAM are not changed.	2.16ms	1.52ms	1.18ms
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S	Assign cursor moving direction and enable the shift of entire display	53μs	38μs	29μs
Display ON/ OFF Control	0	0	0	0	0	0	1	D	C	B	Set display (D), cursor(C), and blinking of cursor(B) on/off control bit.	53μs	38μs	29μs
Cursor or Display Shift	0	0	0	0	0	1	S/C	R/L	-	-	Set cursor moving and display shift control bit, and the direction, without changing of DDRAM data.	53μs	38μs	29μs
Function Set	0	0	0	0	1	DL	N	F	-	-	Set interface data length (DL: 8-bit/4-bit), numbers of display line (N: 2-line/1-line) and, display font type (F:5x10 dots/5x8 dots)	53μs	38μs	29μs
Set CGRAM Address	0	0	0	1	AC5	AC4	AC3	AC2	AC1	AC0	Set CGRAM address in address counter.	53μs	38μs	29μs
Set DDRAM Address	0	0	1	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Set DDRAM address in address counter	53μs	38μs	29μs
Read Busy Flag and Address Counter	0	1	BF	AC6	AC5	AC4	AC3	AC2	AC1	AC0	Whether during internal operation or not can be known by reading BF. The contents of address counter can also be read.			
Write Data to RAM	1	0	D7	D6	D5	D4	D3	D2	D1	D0	Write data into internal RAM (DDRAM/CGRAM).	53μs	38μs	29μs
Read Data from RAM	1	1	D7	D6	D5	D4	D3	D2	D1	D0	Read data from internal RAM (DDRAM/CGRAM).	53μs	38μs	29μs

Note1: "--": don't care

Note2: In the operation condition under -20℃ ~ 75℃, the maximum execution time for majority of instruction sets is 100us, except two instructions, "Clear Display" and "Return Home", in which maximum execution time can take up to 4.1ms.

ORISE Tech Confidential For THREEMEN Use Only

12.0 STANDARD CHARACTER PATTERNS

Upper 4 Bits / Lower 4 Bits	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
xxxx0000	CG RAM (1)															
xxxx0001	(2)															
xxxx0010	(3)															
xxxx0011	(4)															
xxxx0100	(5)															
xxxx0101	(6)															
xxxx0110	(7)															
xxxx0111	(8)															
xxxx1000	(1)															
xxxx1001	(2)															
xxxx1010	(3)															
xxxx1011	(4)															
xxxx1100	(5)															
xxxx1101	(6)															
xxxx1110	(7)															
xxxx1111	(8)															

Note: The character generator RAM is the RAM with which the user can rewrite character patterns by program.

13.0 PRECAUTION FOR USING LCM

1. When design the product with this LCD Module, make sure the viewing angle matches to its purpose of usage.
2. As LCD panel is made of glass substrate, Dropping the LCD module or banging it against hard objects may cause cracking or fragmentation. Especially at corners and edges.
3. Although the polarizer of this LCD Module has the anti-glare coating, always be careful not to scratch its surface. Use of a plastic cover is recommended to protect the surface of polarizer.
4. If the LCD module is stored at below specified temperature, the LC material may freeze and be deteriorated. If it is stored at above specified temperature, the molecular orientation of the LC material may change to Liquid state and it may not revert to its original state. Excessive temperature and humidity could cause polarizer peel off or bubble. Therefore, the LCD module should always be stored within specified temperature range.
5. Saliva or water droplets must be wiped off immediately as those may leave stains or cause color changes if remained for a long time. Water vapor will cause corrosion of ITO electrodes.
6. If the surface of LCD panel needs to be cleaned, wipe it swiftly with cotton or other soft cloth. If it is not still clean enough, blow a breath on the surface and wipe again.
7. The module should be driven according to the specified ratings to avoid malfunction and permanent damage. Applying DC voltage cause a rapid deterioration of LC material. Make sure to apply alternating waveform by continuous application of the M signal. Especially the power ON/OFF sequence should be kept to avoid latch-up of driver LSIs and DC charge up to LCD panel.
8. Mechanical Considerations
   a) 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.
   b) Do not tamper in any way with the tabs on the metal frame.
   c) Do not modify the PCB by drilling extra holes, changing its outline, moving its components or modifying its pattern.
   d) Do not touch the elastomer connector; especially insert a backlight panel (for example, EL).
   e) When mounting a LCM makes sure that the PCB is not under any stress such as bending or twisting. Elastomer contacts are very delicate and missing pixels could result from slight dislocation of any of the elements.
   f) Avoid pressing on the metal bezel, otherwise the elastomer connector could be deformed and lose contact, resulting in missing pixels.
9. Static Electricity
   a) Operator

Ware the electrostatics shielded clothes because human body may be statically charged if not ware shielded clothes. 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.

   b) Equipment

There is a possibility that the static electricity is charged to the equipment, which has a function of peeling or friction action (ex: conveyer, soldering iron, working table). Earth the equipment through proper resistance (electrostatic earth: $1x10^8$ ohm).
Only properly grounded soldering irons should be used.
If an electric screwdriver is used, it should be well grounded and shielded from commutator sparks.
The normal static prevention measures should be observed for work clothes and working benches; for the latter conductive (rubber) mat is recommended.

   c) Floor

Floor is the important part to drain static electricity, which is generated by operators or equipment.

There is a possibility that charged static electricity is not properly drained in case of insulating floor. Set the electrostatic earth (electrostatic earth: $1x10^8$ ohm).

   d) Humidity

Proper humidity helps in reducing the chance of generating electrostatic charges. Humidity should be kept over 50%RH.

   e) Transportation/storage

The storage materials also need to be anti-static treated because there is a possibility that the human body or storage materials such as containers may be statically charged by friction or peeling.

The modules should be kept in antistatic bags or other containers resistant to static for storage.

   f) Soldering

Solder only to the I/O terminals. Use only soldering irons with proper grounding and no leakage.
Soldering temperature : 280 ° C ± 10 ° C

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

g) Others

The laminator (protective film) is attached on the surface of LCD panel to prevent it from scratches or stains. It should be peeled off slowly using static eliminator.

Static eliminator should also be installed to the workbench to prevent LCD module from static charge.

10. Operation
    a) Driving voltage should be kept within specified range; excess voltage shortens display life.
    b) Response time increases with decrease in temperature.
    c) 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”.
    d) Mechanical disturbance during operation (such as pressing on the viewing area) may cause the segments to appear “fractured”.
11. If any fluid leaks out of a damaged glass cell, wash off any human part that comes into contact with soap and water. The toxicity is extremely low but caution should be exercised at all the time.
12. Disassembling the LCD module can cause permanent damage and it should be strictly avoided.
13. LCD retains the display pattern when it is applied for long time (Image retention). To prevent image retention, do not apply the fixed pattern for a long time. Image retention is not a deterioration of LCD. It will be removed after display pattern is changed.
14. Do not use any materials, which emit gas from epoxy resin (hardener for amine) and silicone adhesive agent (dealcohol or deoxym) to prevent discoloration of polarizer due to gas.
15. Avoid the exposure of the module to the direct sunlight or strong ultraviolet light for a long time.

The brightness of LCD module may be affected by the routing of CCFL cables due to leakage to the chassis through coupling effect. The inverter circuit needs to be designed taking the level of leakage current into consideration. Thorough evaluation is needed for LCD module and inverter built into its host equipment to ensure specified brightness.