



*Your Total LCD Solution Provider*

<http://www.orientdisplay.com>

**SPECIFICATION  
FOR  
TFT MODULE**

**MODULE NO: AFA480272DM-4.3-A128**

**REVISION NO: 00**

Customer's Approval:

|  |
|--|
|  |
|--|

|                           | SIGNATURE | DATE |
|---------------------------|-----------|------|
| PREPARED BY (RD ENGINEER) |           |      |
| CHECKED BY                |           |      |
| APPROVED BY               |           |      |

# Table of Contents

|      |                            |    |
|------|----------------------------|----|
| 1.0  | General Description        | 3  |
| 2.0  | Electrical Characteristics | 3  |
| 3.0  | Mechanical Parameters      | 3  |
| 4.0  | Pin Definition             | 3  |
| 5.0  | Interface Design           | 4  |
| 6.0  | Control Command            | 4  |
| 7.0  | Outline Drawing            | 8  |
| 8.0  | User's Guide               | 9  |
| 9.0  | Reliability Test Items     | 11 |
| 10.0 | Package Specification      | 11 |

M3 TFT is target to the users updating their product from monochrome LCD to TFT displays or new products fast developing; our solution can provides a simple and reliable way to control TFT displays. M3-M series include small screen with RGB port from 3.5" to 8.0", meeting the requirements for various applications from customers.

## 1.0 General Description

1. Built-in command table
2. Typical data space 128M Byte
3. Communication interface Uart (Customizable RS232)
4. Data updated using USB port
5. Font 8\*16、12\*24、16\*32、24\*48、32\*64 or own definition

## 2.0 Electrical Parameters

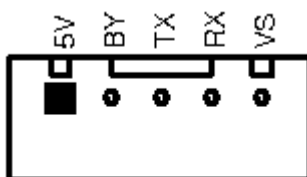
|                       |                             |
|-----------------------|-----------------------------|
| <b>Name</b>           | <b>AFA480272DM-4.3-A128</b> |
| Input voltage         | 5V±5%                       |
| Current               | <250mA                      |
| Color Depth           | 65K colors                  |
| Operating Temperature | -20---70℃                   |
| Storage Temperature   | -30----80℃                  |
| Luminance             | 520cd/m <sup>2</sup>        |
| Display Mode          | Transmissive                |
| Resolution            | 480*272                     |

## 3.0 Mechanical Parameters (mm)

|                 |                             |
|-----------------|-----------------------------|
| <b>ITEM</b>     | <b>AFA480272DM-4.3-A128</b> |
| Size of panel   | 105.50*67.20                |
| Size of module  | 122.00*74.70*13.9           |
| Size (VA) Bezel | 98.10*57.00                 |

## 4.0 Pin Definition

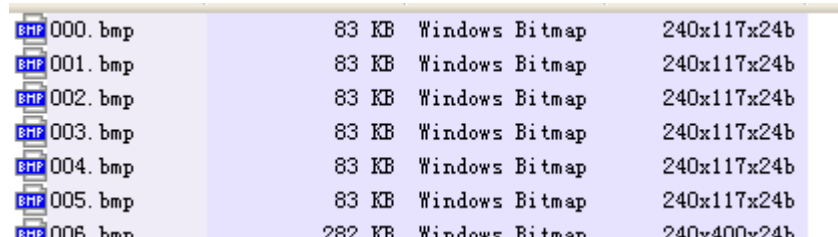
J1 on module is communication port and K1 is reset key.



## 5.0 Interface Design

1. Firstly, put all pictures needed into the folder of BMP\_FILE, convert them into BMP format and change picture names in proper sequence from 000.BMP. The operation can be completed with software of ACDSEE; see [Appendix I](#) for details in operation.

The picture files with names in proper sequence are shown in figure below:



|     |          |        |                |             |
|-----|----------|--------|----------------|-------------|
| BMP | 000. bmp | 83 KB  | Windows Bitmap | 240x117x24b |
| BMP | 001. bmp | 83 KB  | Windows Bitmap | 240x117x24b |
| BMP | 002. bmp | 83 KB  | Windows Bitmap | 240x117x24b |
| BMP | 003. bmp | 83 KB  | Windows Bitmap | 240x117x24b |
| BMP | 004. bmp | 83 KB  | Windows Bitmap | 240x117x24b |
| BMP | 005. bmp | 83 KB  | Windows Bitmap | 240x117x24b |
| BMP | 006. bmp | 282 KB | Windows Bitmap | 240x400x24b |

2. Execute batch processing command Bmptobin.exe and two files will be generated in the folder of BMP\_DATA after execution, among which BMPDATA.BIN is the data file where all picture data are put successively and TABLE.BIN is the table file equivalent to index file; copy these two files into the folder virtualized by the module and then they are available for use.

3. If special character display is required by user, like Chinese or Japanese etc. The software attached in this module can be used for extracting font library; ([Only for Chinese version](#))

Revise the name of extracted font library file and copy it in to USB flash disk for application; the file name should be changed as HZK.BIN.

4. Note: this USB flash disk is a virtualized one; the file system is regulated for the convenience of application of this module and improvement of picture refreshing speed; please use FAT12 to format the USB flash disk; the limit number of files in root directory is 16 and generally there are only font library file, table file and data file; if other files need to be stored by user, it had better to make a new folder for storage of these files.

5. Press K1 key (reset key) on module after all files are copied into USB flash disk and then they are available for use.

## 6.0 Control Command

This module adopts alphabetic string to control all operations.

The command format begins with command word continued with parameter list in which parameters are separated by space and ends with a return character; pay attention that the return character is a must. When the command is executing, the busy pin is low, and the module can't receive new command, until the busy pin is high.

"CMD n1 n2.....Return"

The parameters n1, n2... in table below express a 16-bit unsigned integer data and s1, s2...expresses alphabetic string.

## Drawing commands

| Function                                 | Command Format               | Example                     |                                |
|--|------------------------------|-----------------------------|--------------------------------|
| Browse Pictures                          | ALL                          | "ALL\n"                     |                                |
| Draw a circle                            | CIRCLE Xa Ya R C             | "CIRCLE 100 100 50 31\n"    |                                |
| Fill in color                            | CLR Xa Ya Xe Ye C            | "CLR 0 0 100 100 31\n"      |                                |
| Clear screen                             | CLS C                        | "CLS 31\n"                  |                                |
| Cut a picture                            | CUT Pn Xa Ya Xb Yb Xs Ys     | "CUT 1 30 30 0 0 100 100\n" |                                |
| Draw a dot                               | DOT Xa Ya C                  | "DOT 100 100 31\n"          |                                |
| Draw a line                              | LINE Xa Ya Xe Ye C           | "LINE 10 10 50 50 31\n"     |                                |
| Backlight on                             | LEDON                        | "LEDON\n"                   |                                |
| Adjusting backlight luminance            | LEDON X (X:1~10)             | "LEDON10\n"                 |                                |
| Backlight off                            | LEDOFF                       | "LEDOFF\n"                  |                                |
| LOAD PIC                                 | PIC Pn Xa Ya                 | "PIC 1 30 30\n"             |                                |
| Display image files directly             | LOAD Xa Ya Path              | "LOAD 0 0 jpg\jpg03.jpg\n"  | Support<br>BMP,JPG,TGA,PN<br>G |
| Draw a rectangle                         | RECT Xa Ya Xe Ye C           | "RECT 10 10 100 100 31\n"   |                                |
| *Get screen size                         | SIZE                         | "SIZE\n"                    |                                |
| Display alphabetic string                | STR Xa Ya C Str              | "STR 0 0 31 ABCDEF\n"       |                                |
| Display alphabetic string with grounding | BSTR Fontsize Xa Ya C Cb Str | "BSTR 16 0 0 31 0 ABCDEF\n" |                                |
| Display font user-defined                | UF FontIndex Xa Ya C Str     | "UF 0 0 0 31 ABCDEF\n"      |                                |
| Display font with grounding user-defined | BUF FontIndex Xa Ya C Cb Str | "BUF 0 0 0 31 0 ABCDEF\n"   |                                |
| Buzzer control                           | BUZ Tb Fb                    | "BUZ 300 2000\n"            |                                |

## Display alphabetic string. Copy the font file to flash before use these commands

| Function                           | Command Format   | Example                 |                 |
|------------------------------------|------------------|-------------------------|-----------------|
| Display alphabetic string(Font 16) | HZ16 Xa Ya C Str | " HZ16 0 0 31 ABCDEF\n" | Font Size 8x16  |
| Display alphabetic string(Font 24) | HZ24 Xa Ya C Str | " HZ24 0 0 31 ABCDEF\n" | Font Size 12x24 |
| Display alphabetic string(Font 32) | HZ32 Xa Ya C Str | " HZ32 0 0 31 ABCDEF\n" | Font Size 16x32 |
| Display alphabetic string(Font 48) | HZ48 Xa Ya C Str | " HZ48 0 0 31 ABCDEF\n" | Font Size 24x48 |

|  |                      |                            |                 |
|--|----------------------|----------------------------|-----------------|
| Display alphabetic string(Font 64)                 | HZ64 Xa Ya C Str     | " HZ64 0 0 31 ABCDEF\n"    | Font Size 32x64 |
| Display alphabetic string with back color(Font 16) | HZB16 Xa Ya C Cb Str | " HZB16 0 0 31 0 ABCDEF\n" | Font Size 8x16  |
| Display alphabetic string with back color(Font 24) | HZB24 Xa Ya C Cb Str | " HZB24 0 0 31 0 ABCDEF\n" | Font Size 12x24 |
| Display alphabetic string with back color(Font 32) | HZB32 Xa Ya C Cb Str | " HZB32 0 0 31 0 ABCDEF\n" | Font Size 16x32 |
| Display alphabetic string with back color(Font 48) | HZB48 Xa Ya C Cb Str | " HZB48 0 0 31 0 ABCDEF\n" | Font Size 24x48 |
| Display alphabetic string with back color(Font 64) | HZB64 Xa Ya C Cb Str | " HZB64 0 0 31 0 ABCDEF\n" | Font Size 32x64 |

### Animation, the series picture must be same size \*\*

| Function           | Command Format           | Example               |  |
|--------------------|--------------------------|-----------------------|--|
| <b>*Animation1</b> | M1 On/Off Xa Ya Ps Pe Pt | "M1 1 0 0 0 8 100\n"  |  |
| <b>Animation2</b>  | M2 On/Off Xa Ya Ps Pe Pt | "M2 1 0 0 0 8 100 \n" |  |
| <b>Animation3</b>  | M3 On/Off Xa Ya Ps Pe Pt | "M3 1 0 0 0 8 100 \n" |  |
| <b>Animation4</b>  | M4 On/Off Xa Ya Ps Pe Pt | "M4 1 0 0 0 8 100 \n" |  |

#### Note:

Xa Ya :start x y coordinates  
C :color(16bits,RGB 565 )

Xe Ye :End x y coordinates  
Cb color underpainting

Xb Yb : start end X Y coordinates in livelong PIC

Xs Ys : Need to intercept the size from livelong picture

Str :alphabetic string(8x16)

Pn : Pictures serial number(000-999)

R : Radius

Ps : Start animation picture number

Pe : End animation picture number

Pt : Spacing time (Unit ms)

Bt : Buzzer on time (Unit ms)

Bf : Buzzer frequency(Unit Hz) 1K-4K

Path Denote the path of the picture file BMP,JPG,PNG,TGA

Fontsize 8\*16, 12\*24, 16\*32, 24\*48, 32\*64

FontIndex user-defined font index (0~19)

\*\* Please ensure the command will be ended with the HEX value of " 0x0d & 0x0a".

If "LINE 10 10 50 50 31 \n" doesn't work, please try "LINE 10 10 50 50 31 \nr" or "LINE

10

10 50 50 31\r\n"

("\\n": 0x0d & 0x0a.)

**\*\*Parameter configuration:**

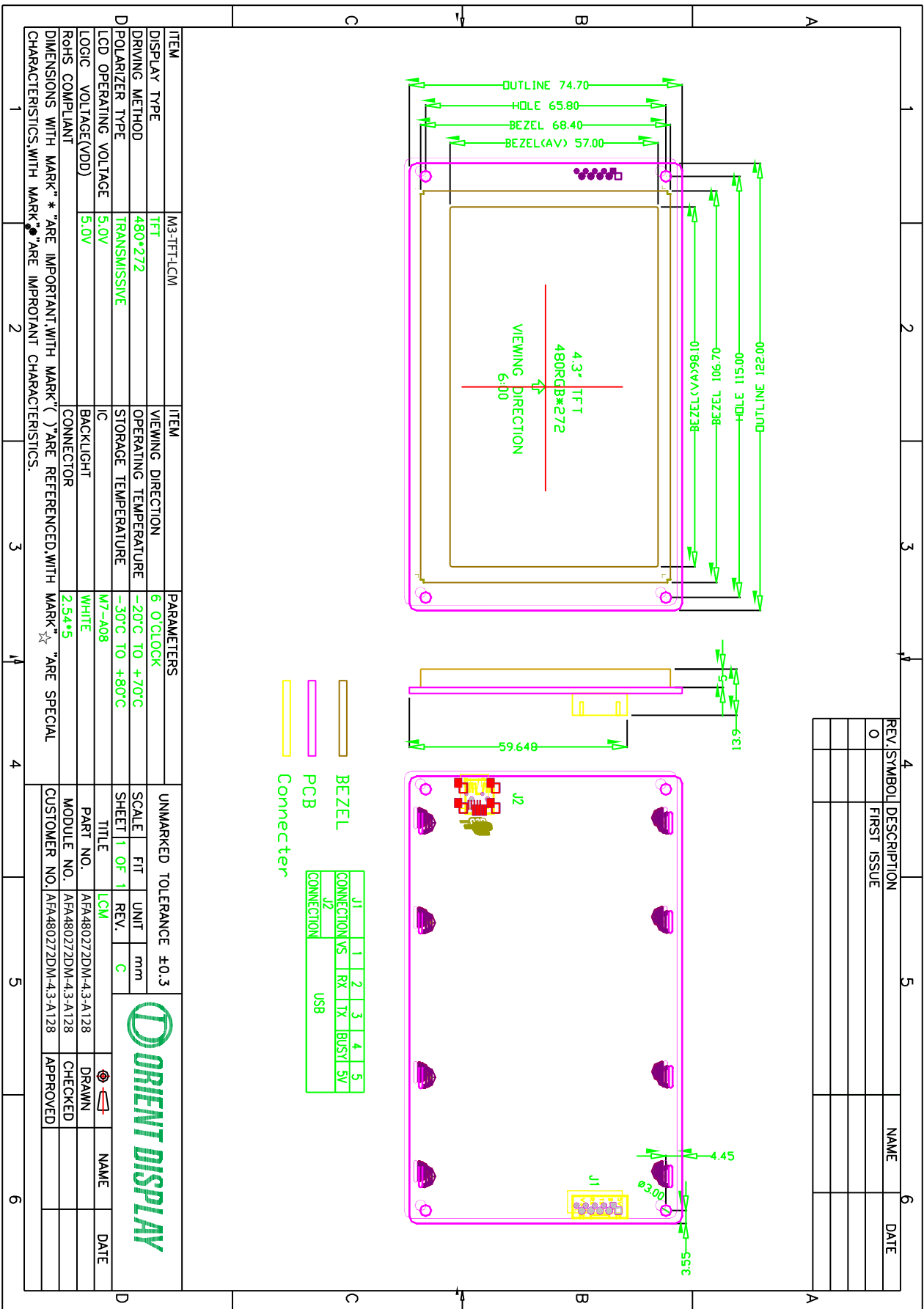
This part use config.ini to set module basic parameters like UART baud rate, touch panel calibrating, demo setting, etc...

Config.ini already existed in flash memory before shipping out.

Below is the comments for config.ini

```
BaudRate=9600 // Uart baud rate
StartColor=30 // The color of starting
DemoStatus=1 // Demo ON (or OFF)
DemoTime=1000 // Demo interval (ms)
DemoXaddr=0 // Demo pic coordinate X
DemoYaddr=0 // Demo pic coordinate Y
Monitor=0 // Uart monitor
MotionSize1=1024 //Motion picture size (KB)
MotionSize2=200
MotionSize3=0
MotionSize4=0
TpTestFlag=1 // Test TP, it will display a small circle at clicking position.
TpColor=63488 // Color of small circle
TpInterval=10 // Interval of TP testing (ms)
***** DO NOT modify below KEY TFT timing parameters *****
TFT_CLK=0 //
TFT_THD= 800 //
TFT_THPW=48 //
TFT_THB=40 //
TFT_THFP=40 //
TFT_TVD=480 //
TFT_TVPW=3 //
TFT_TVB=29 //
TFT_TVFP=13 //
-----
RTPHV=0 //RTP wires order
XA=350
XB=-13605
YA=259
YB=-38457
```

# 7.0 Outline Drawing





# 8.0 User's Guide

## Appendix I

### Application of ACDSEE for classifying pictures

1. Collect materials for use into the folder of BMP\_FILE; at the beginning, the pictures may be disordered and the processing of following steps is needed to make them in proper order for operation.



2. Convert all files into BMP format.



3. Sort the pictures successively beginning with 000.



4. Reorder the special picture sequence, for instance, the animation part and special font part must be continuous.

5. Execute the batch file of BmptoBin.exe

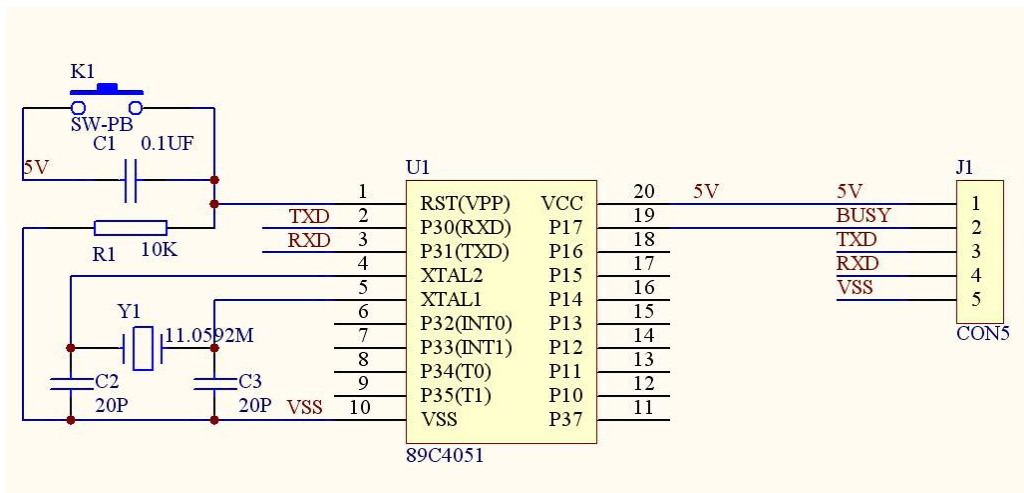


6. After execution, there are two files generated in BMP\_DATA, wherein BMPDATA.BIN is data file in which all picture data are put inside in order; TABLE.BIN is table file equivalent to index file. Copy these two files into the folder virtualized the module and then they are available for use.



## Appendix II

### Application circuit



## 9.0 Reliability Test Items

| No | Item                       | Condition                            | Quantity |
|----|----------------------------|--------------------------------------|----------|
| 1  | High Temperature Operating | 50°C, 96Hrs                          | 5        |
| 2  | Low Temperature Operating  | 0°C, 96Hrs                           | 5        |
| 3  | High Humidity              | 60°C, 90%RH, 96Hrs                   | 5        |
| 4  | High Temperature Storage   | 80°C, 96Hrs                          | 5        |
| 5  | Low Temperature Storage    | -40°C, 96Hrs                         | 5        |
| 6  | Thermal shock              | -20°C, 30min~70°C, 30min, 20 cycles. | 5        |

Note1. No deflection cosmetic and operational function allowable.

## 10.0 Package Specification

TBD