Introduction

3.2 inch 320*240 DOTS Multicolor Graphic LCD, with touch screen and stand-alone controllers

More

Resources

- User Manual
- Schematic
- Dimension
- Code
- Datasheets

FAQ

Question:
Parameters of the LCD?

Answer: [Collapse]

- LCD power supply: 2.5V~3.3V
- Back light power supply: 4.5V~5V
- Average working current of the back light: 150mA
- Average working current of the LCD panel: 15mA

Question:
What's the difference between 3.2inch 320x240 Touch LCD (C) and the older 3.2inch 320x240 Touch LCD (A)?

Answer: [Collapse]

The same:

- Pin mapping and resolution are the same.

The difference:

- The LCD controller of 3.2inch 320x240 Touch LCD (A) is SSD1289
- The LCD controller of 3.2inch 320x240 Touch LCD (C) is ILI9325

Question:

Is the program for type (A) compatible with type (C)?

Answer: [Collapse]

Although the pin mapping is the same, you have to use the newer program for type (C) LCDs.

Question:

The LCD shows abnormal image, how to solve this problem?

Answer: [Collapse]

- The Grmma value may differ depending on batch, so the initialization statements of the demo may be unsuitable for some LCDs. You can try the following init code:

```
if( DeviceCode == 0x9325 || DeviceCode == 0x9328 )
 {
        LCD_Code = ILI9325;
        LCD_WriteReg(0xE5, 0x78F0); /* set SRAM internal timing */
        LCD_WriteReg(0x01, 0x0100); /* set Driver Output Control */
        LCD_WriteReg(0x02, 0x0700); /* set 1 line inversion */
        LCD_WriteReg(0x03, 0x1030); /* set GRAM write direction and BGR=1 */
        LCD_WriteReg(0x04, 0x0000); /* Resize register */
        LCD_WriteReg(0x08, 0x0207); /* set the back porch and front porch */
        LCD_WriteReg(0x09, 0x0000); /* set non-display area refresh cycle ISC[3:0] */
        LCD_WriteReg(0x0A, 0x0000); /* FMARK function */
        LCD_WriteReg(0x0C, 0x0000); /* RGB interface setting */
        LCD_WriteReg(0x0D, 0x0000); /* Frame marker Position */
        LCD_WriteReg(0x0F, 0x0000); /* RGB interface polarity */
        /*************Power On sequence ****************/
        LCD_WriteReg(0x10, 0x0000); /* SAP, BT[3:0], AP, DSTB, SLP, STB */
        LCD_WriteReg(0x11, 0x0007); /* DC1[2:0], DC0[2:0], VC[2:0] */
        LCD_WriteReg(0x12, 0x0000); /* VREG1OUT voltage */
        LCD_WriteReg(0x13, 0x0000); /* VDV[4:0] for VCOM amplitude */
        LCD_WriteReg(0x07, 0x0001);
        delay_ms(200);
        /* Dis-charge capacitor power voltage */
        LCD_WriteReg(0x10, 0x1090); /* SAP, BT[3:0], AP, DSTB, SLP, STB */
        LCD_WriteReg(0x11, 0x0227); /* Set DC1[2:0], DC0[2:0], VC[2:0] */
        delay_ms(50);                           /* Delay 50ms */
        LCD_WriteReg(0x12, 0x001F);
        delay_ms(50);                           /* Delay 50ms */
        LCD_WriteReg(0x13, 0x1500); /* VDV[4:0] for VCOM amplitude */
        LCD_WriteReg(0x29, 0x0027); /* 04 VCM[5:0] for VCOMH */
        LCD_WriteReg(0x2B, 0x000D); /* Set Frame Rate */
        delay_ms(50);                           /* Delay 50ms */
        LCD_WriteReg(0x20, 0x0000); /* GRAM horizontal Address */
        LCD_WriteReg(0x21, 0x0000); /* GRAM Vertical Address */
        /* ----------- Adjust the Gamma Curve ---------- */
#if 1   /* the Gamma value may differ depending on batch, there are 3 groups for reference */
        LCD_WriteReg(0x30, 0x0000);
        LCD_WriteReg(0x31, 0x0603);
        LCD_WriteReg(0x32, 0x0206);
        LCD_WriteReg(0x35, 0x0206);
        LCD_WriteReg(0x36, 0x0004);
        LCD_WriteReg(0x37, 0x0105);
        LCD_WriteReg(0x38, 0x0401);
        LCD_WriteReg(0x39, 0x0707);
        LCD_WriteReg(0x3C, 0x0602);
        LCD_WriteReg(0x3D, 0x0004);
#endif
#if 0
        LCD_WriteReg(0x30, 0x0000);
        LCD_WriteReg(0x31, 0x0707);
        LCD_WriteReg(0x32, 0x0307);
        LCD_WriteReg(0x35, 0x0200);
```

```
        LCD_WriteReg(0x36, 0x0008);
        LCD_WriteReg(0x37, 0x0004);
        LCD_WriteReg(0x38, 0x0000);
        LCD_WriteReg(0x39, 0x0707);
        LCD_WriteReg(0x3C, 0x0002);
        LCD_WriteReg(0x3D, 0x1D04);
#endif
#if 0
        LCD_WriteReg(0x30,0x0007);
        LCD_WriteReg(0x31,0x0707);
        LCD_WriteReg(0x32,0x0006);
        LCD_WriteReg(0x35,0x0704);
        LCD_WriteReg(0x36,0x1f04);
        LCD_WriteReg(0x37,0x0004);
        LCD_WriteReg(0x38,0x0000);
        LCD_WriteReg(0x39,0x0706);
        LCD_WriteReg(0x3c,0x0701);
        LCD_WriteReg(0x3d,0x000f);
#endif
        /* ------------------ Set GRAM area --------------- */
        LCD_WriteReg(0x50, 0x0000); /* Horizontal GRAM Start Address */
        LCD_WriteReg(0x51, 0x00EF); /* Horizontal GRAM End Address */
        LCD_WriteReg(0x52, 0x0000); /* Vertical GRAM Start Address */
        LCD_WriteReg(0x53, 0x013F); /* Vertical GRAM Start Address */
        LCD_WriteReg(0x60, 0xA700); /* Gate Scan Line */
        LCD_WriteReg(0x61, 0x0001); /* NDL,VLE, REV */
        LCD_WriteReg(0x6A, 0x0000); /* set scrolling line */
        /* -------------- Partial Display Control --------- */
        LCD_WriteReg(0x80, 0x0000);
        LCD_WriteReg(0x81, 0x0000);
        LCD_WriteReg(0x82, 0x0000);
        LCD_WriteReg(0x83, 0x0000);
        LCD_WriteReg(0x84, 0x0000);
        LCD_WriteReg(0x85, 0x0000);
        /* -------------- Panel Control ------------------- */
        LCD_WriteReg(0x90, 0x0010);
        LCD_WriteReg(0x92, 0x0600);
        LCD_WriteReg(0x07, 0x0133); /* 262K color and display ON */
}
```