

**a-Si TFT LCD Single Chip Driver with
240RGBx320 Resolution and 262K color**

Application Notes

Version: Preliminary V0.92

Date: April., 21st 2007

ILI TECHNOLOGY CORP.

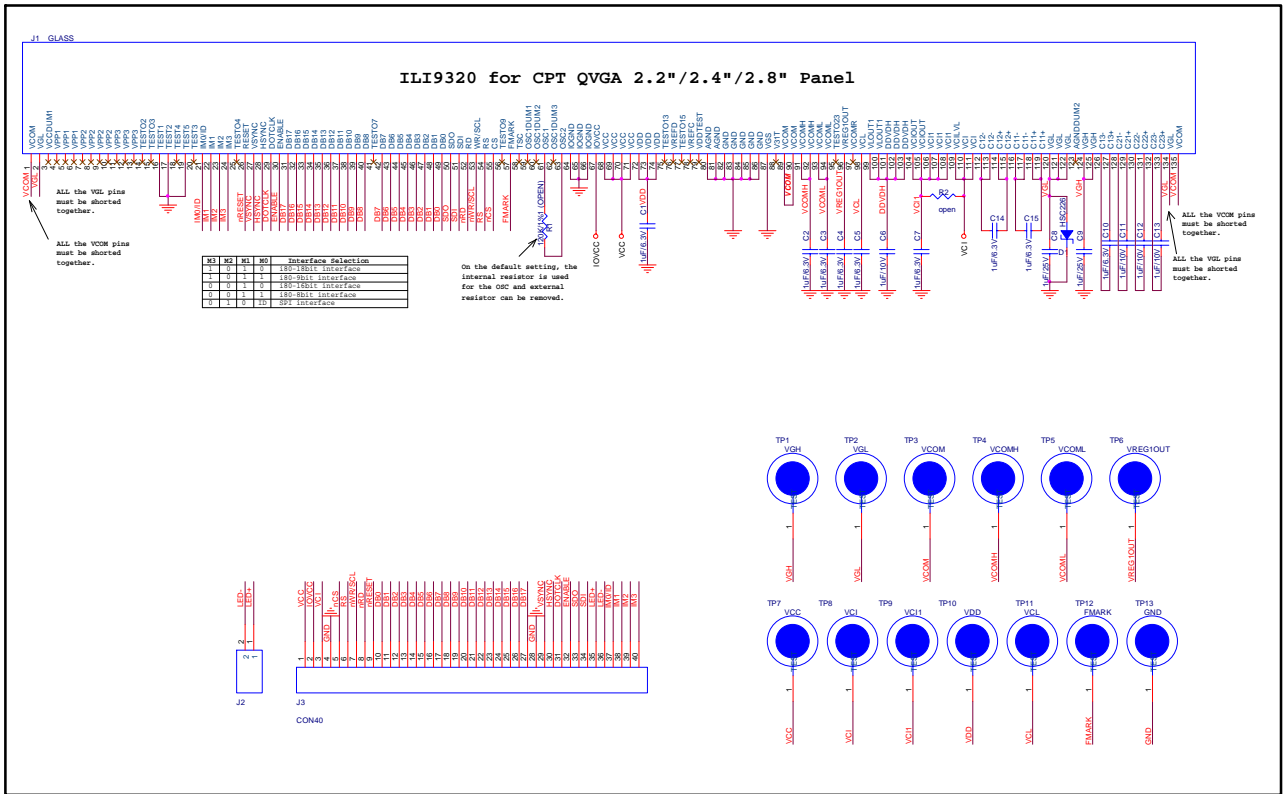
4F, No. 2, Tech. 5th Rd., Hsinchu Science Park,
Taiwan 300, R.O.C.

Tel.886-3-5670095; Fax.886-3-5670096

<http://www.ilitek.com>

1. Application FPC Circuit

1.1. CPT 2.4" & 2.8" Panel



1.1.1. CPT 2.4” Initial Code

void ILI9320_CPT24_Initial(void)

```
{
// VCI=2.8V
//***** Reset LCD Driver *****//
LCD_nRESET = 1;
    delaysms(1); // Delay 1ms
LCD_nRESET = 0;
    delaysms(10); // Delay 10ms           // This delay time is necessary
LCD_nRESET = 1;
    delaysms(50); // Delay 50 ms

//***** Start Initial Sequence *****//
LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the Vcore voltage and this setting is must.
LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
LCD_CtrlWrite_ILI9320(0x0003, 0x1030); // set GRAM write direction and BGR=1.
LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register

LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity

//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0007); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delaysms(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // R11h=0x0031 at VCI=3.3V DC1[2:0], DC0[2:0], VC[2:0]
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013C); // R12h=0x0138 at VCI=3.3V VREG1OUT voltage
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1C00); // R13h=0x1800 at VCI=3.3V VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000E); // R29h=0x0008 at VCI=3.3V VCM[4:0] for VCOMH
    delaysms(50);
LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address
LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address

// ----- Adjust the Gamma Curve -----//
LCD_CtrlWrite_ILI9320(0x0030, 0x0000);
LCD_CtrlWrite_ILI9320(0x0031, 0x0505);
LCD_CtrlWrite_ILI9320(0x0032, 0x0004);

LCD_CtrlWrite_ILI9320(0x0035, 0x0006);
LCD_CtrlWrite_ILI9320(0x0036, 0x0707);
LCD_CtrlWrite_ILI9320(0x0037, 0x0105);
LCD_CtrlWrite_ILI9320(0x0038, 0x0002);
LCD_CtrlWrite_ILI9320(0x0039, 0x0707);

LCD_CtrlWrite_ILI9320(0x003C, 0x0704);
LCD_CtrlWrite_ILI9320(0x003D, 0x0807);
}
```

```
//----- Set GRAM area -----//
LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address

LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // NDL, VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line

//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);

//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

void LCD_ExitSleep_ILI9320(void)
{
//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // R11h=0x0031 at VCI=3.3V DC1[2:0], DC0[2:0], VC[2:0]
    delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013C); // R12h=0x0138 at VCI=3.3V VREG1OUT voltage
    delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1C00); // R13h=0x1800 at VCI=3.3V VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000E); // R29h=0x0008 at VCI=3.3V VCM[4:0] for VCOMH
    delays(50);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

void LCD_EnterSleep_ILI9320(void)
{
LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF
//***** Power OFF sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP
}

```

1.1.2. CPT 2.8” Initial Code

void ILI9320_CPT28_Initial(void)

```
{
// VCI=2.8V
//***** Reset LCD Driver *****/
LCD_nRESET = 1;
    delaysms(1); // Delay 1ms
LCD_nRESET = 0;
    delaysms(10); // Delay 10ms
LCD_nRESET = 1;
    delaysms(50); // Delay 50 ms

//***** Start Initial Sequence *****/
    LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the internal vcore voltage
LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
LCD_CtrlWrite_ILI9320(0x0003, 0x1030); // set GRAM write direction and BGR=1.
LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register

LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity

//*****Power On sequence *****/
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0007); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delaysms(200); // Dis-charge capacitor power voltage

LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013A); // VREG1OUT voltage
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1600); // VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000C); // VCM[4:0] for VCOMH
    delaysms(50);

LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address
LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address

// ----- Adjust the Gamma Curve -----//
LCD_CtrlWrite_ILI9320(0x0030, 0x0504);
LCD_CtrlWrite_ILI9320(0x0031, 0x0703);
LCD_CtrlWrite_ILI9320(0x0032, 0x0702);
LCD_CtrlWrite_ILI9320(0x0035, 0x0101);
LCD_CtrlWrite_ILI9320(0x0036, 0x0A1F);
LCD_CtrlWrite_ILI9320(0x0037, 0x0504);
LCD_CtrlWrite_ILI9320(0x0038, 0x0003);
LCD_CtrlWrite_ILI9320(0x0039, 0x0706);
LCD_CtrlWrite_ILI9320(0x003C, 0x0707);
LCD_CtrlWrite_ILI9320(0x003D, 0x091F);
}
```

```
//----- Set GRAM area -----//
LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address

LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // NDL, VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line

//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);

//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);

LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}
```

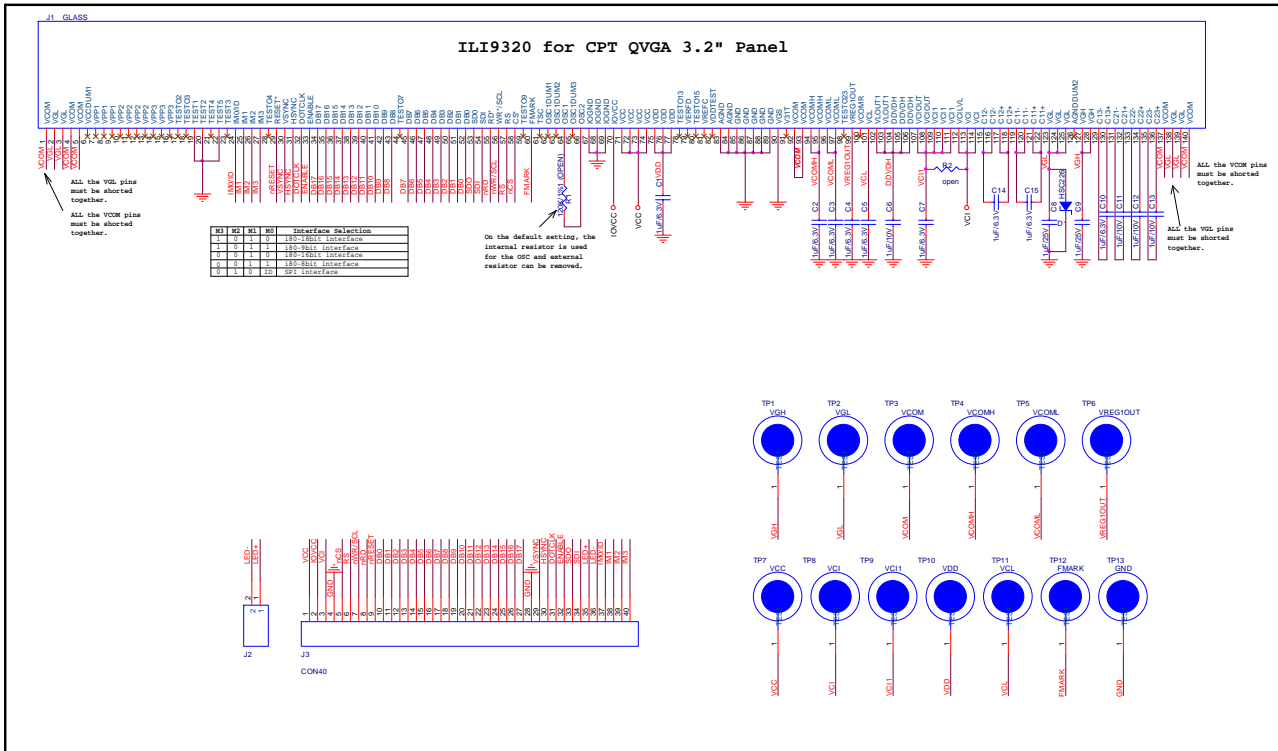
void LCD_ExitSleep_ILI9320(void)

```
{
//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013A); // VREG1OUT voltage
delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1600); //VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000C); //VCM[4:0] for VCOMH
delays(50);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}
```

void LCD_EnterSleep_ILI9320(void)

```
{
LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF
//***** Power OFF sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP
}
```

1.2. CPT3.2" Panel



1.2.1. CPT 3.2" Initial Code

void ILI9320_CPT32_Initial(void)

```
{
// VCI=2.8V
//***** Reset LCD Driver *****//
    LCD_nRESET = 1;
        delaysms(1); // Delay 1ms
    LCD_nRESET = 0;
        delaysms(10); // Delay 10ms
    LCD_nRESET = 1;
        delaysms(50); // Delay 50 ms

//***** Start Initial Sequence *****//

    LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the internal vcore voltage
    LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
    LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
    LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
    LCD_CtrlWrite_ILI9320(0x0003, 0x1030); // set GRAM write direction and BGR=1.
    LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register
    LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
    LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
    LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
    LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
    LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
    LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity

//*****Power On sequence *****//
    LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0007); // DC1[2:0], DC0[2:0], VC[2:0]
    LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
    LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
        delaysms(200); // Dis-charge capacitor power voltage
    LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0147); // DC1[2:0], DC0[2:0], VC[2:0]
        delaysms(50); // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0012, 0x013C); // VREG1OUT voltage
        delaysms(50); // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0013, 0x0E00); // VDV[4:0] for VCOM amplitude
    LCD_CtrlWrite_ILI9320(0x0029, 0x0009); // VCM[4:0] for VCOMH
        delaysms(50);
    LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address
    LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address

// ----- Adjust the Gamma Curve -----//
    LCD_CtrlWrite_ILI9320(0x0030, 0x0207);
    LCD_CtrlWrite_ILI9320(0x0031, 0x0505);
    LCD_CtrlWrite_ILI9320(0x0032, 0x0102);
    LCD_CtrlWrite_ILI9320(0x0035, 0x0006);
    LCD_CtrlWrite_ILI9320(0x0036, 0x0606);
    LCD_CtrlWrite_ILI9320(0x0037, 0x0707);
    LCD_CtrlWrite_ILI9320(0x0038, 0x0506);
    LCD_CtrlWrite_ILI9320(0x0039, 0x0407);
    LCD_CtrlWrite_ILI9320(0x003C, 0x0106);
    LCD_CtrlWrite_ILI9320(0x003D, 0x0601);
//----- Set GRAM area -----//
    LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
    LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
    LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
}
```



```

LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address

LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // NDL,VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line

//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);

//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);

LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

```

void LCD_ExitSleep_ILI9320(void)

```

{
//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
delayms(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0147); // DC1[2:0], DC0[2:0], VC[2:0]
delayms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013C); // VREG1OUT voltage
delayms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x0E00); // VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x0009); // VCM[4:0] for VCOMH
delayms(50);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

```

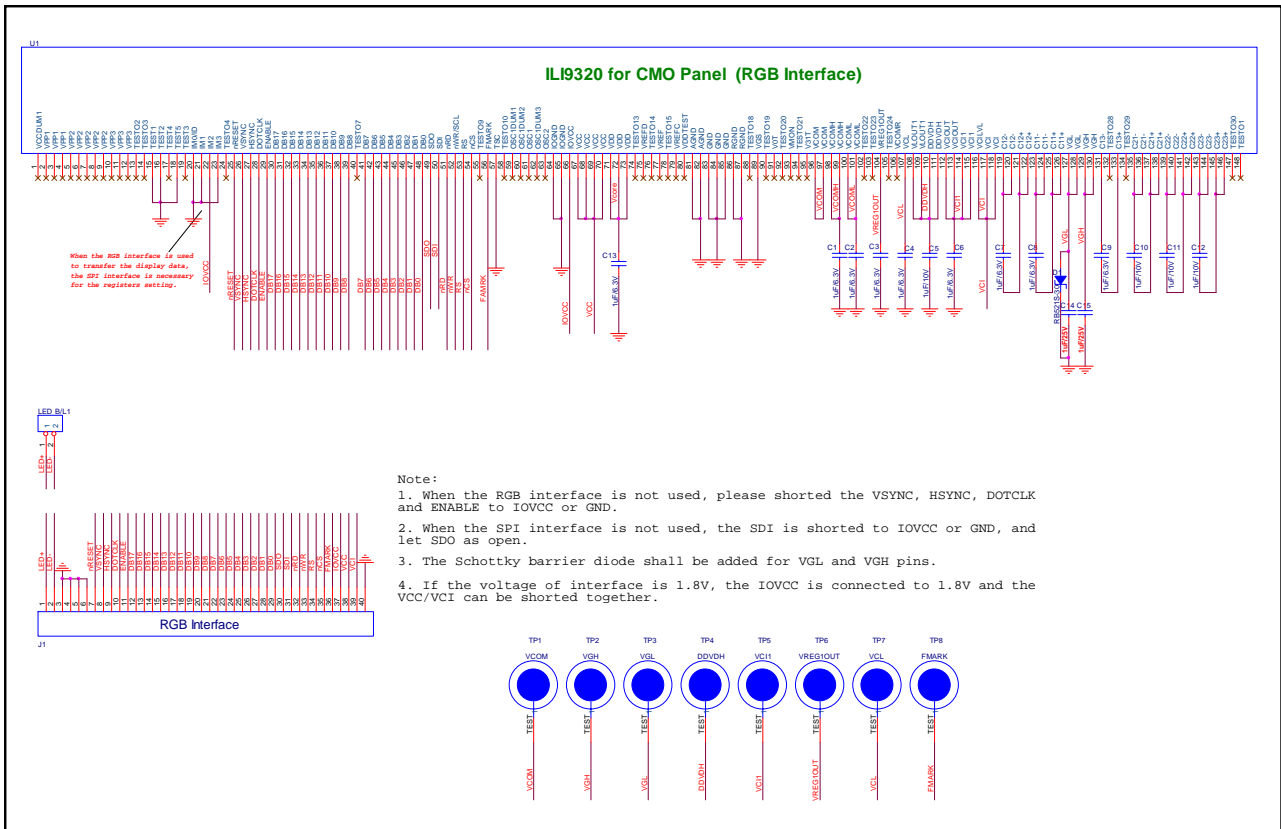
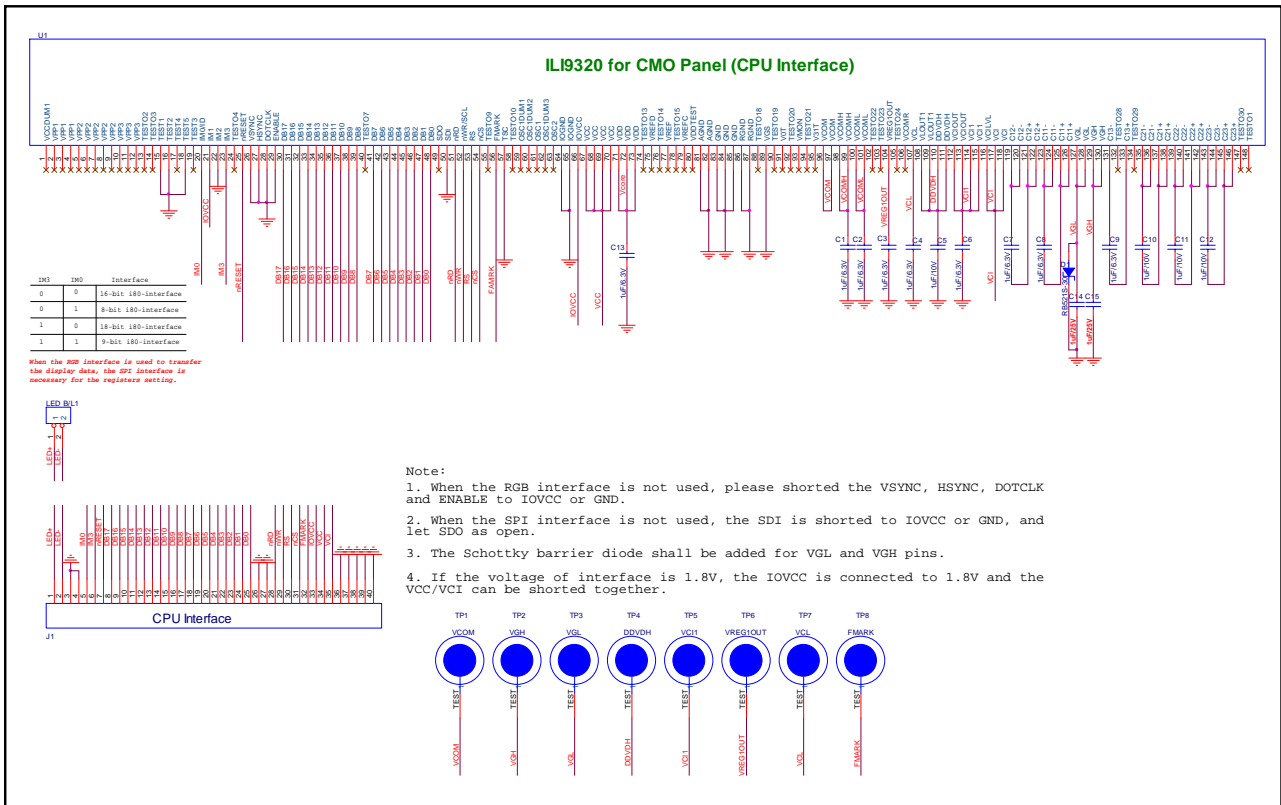
void LCD_EnterSleep_ILI9320(void)

```

{
LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF
//***** Power OFF sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
delayms(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP
}

```

1.3. CMO 2.2" & CMO 2.4" & CMO 2.8" Panel



1.3.1. CMO 2.2" initial code

void ILI9320_CMO22_Initial(void)

```
{
    // VCI=2.8V
    //***** Reset LCD Driver *****//
    LCD_nRESET = 1;
    delayms(1); // Delay 1ms
    LCD_nRESET = 0;
    delayms(10); // Delay 10ms           // This delay is necessary
    LCD_nRESET = 1;
    delayms(50); // Delay 50 ms

    //***** Start Initial Sequence *****//

    LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the Vcore voltage and this setting is must.
    LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
    LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
    LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
    LCD_CtrlWrite_ILI9320(0x0003, 0x0030); // set GRAM write direction and BGR=0.
    LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register

    LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
    LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
    LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
    LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
    LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
    LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity

    //*****Power On sequence *****//
    LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0007); // DC1[2:0], DC0[2:0], VC[2:0]
    LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
    LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delayms(200); // Dis-charge capacitor power voltage
    LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
    delayms(50); // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0012, 0x013E); // VREG1OUT voltage
    delayms(50); // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0013, 0x1C00); // VDV[4:0] for VCOM amplitude
    LCD_CtrlWrite_ILI9320(0x0029, 0x0007); // VCM[4:0] for VCOMH
    delayms(50);
    LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address
    LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address

    // ----- Adjust the Gamma Curve -----//
    LCD_CtrlWrite_ILI9320(0x0030, 0x0000);
    LCD_CtrlWrite_ILI9320(0x0031, 0x0207);
    LCD_CtrlWrite_ILI9320(0x0032, 0x0707);

    LCD_CtrlWrite_ILI9320(0x0035, 0x0203);
    LCD_CtrlWrite_ILI9320(0x0036, 0x0609);
    LCD_CtrlWrite_ILI9320(0x0037, 0x0707);
    LCD_CtrlWrite_ILI9320(0x0038, 0x0400);
    LCD_CtrlWrite_ILI9320(0x0039, 0x0707);
    LCD_CtrlWrite_ILI9320(0x003C, 0x0500);
    LCD_CtrlWrite_ILI9320(0x003D, 0x1807);
}
```

```
//----- Set GRAM area -----//
LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address

LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // NDL, VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line

//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);

//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}
```

void LCD_ExitSleep_ILI9320(void)

```
{
//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
    delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013E); // VREG1OUT voltage
    delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1C00); // VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x0007); // VCM[4:0] for VCOMH
    delays(50);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}
```

void LCD_EnterSleep_ILI9320(void)

```
{
LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF

//***** Power OFF sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP
}
```

1.3.1. CMO 2.4" initial code

void ILI9320_CMO24_Initial(void)

```
{
// VCI=2.8V
//***** Reset LCD Driver *****//
LCD_nRESET = 1;
    delaysms(1); // Delay 1ms
LCD_nRESET = 0;
    delaysms(10); // Delay 10ms           // This delay time is necessary
LCD_nRESET = 1;
    delaysms(50); // Delay 50 ms

//***** Start Initial Sequence *****//
LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the internal vcore voltage
LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
LCD_CtrlWrite_ILI9320(0x0003, 0x1030); // set GRAM write direction and BGR=1.
LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register

LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity

//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delaysms(200); // Dis-charge capacitor power voltage

LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013B); // VREG1OUT voltage
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1900); // VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x0007); // VCM[4:0] for VCOMH
    delaysms(50);
LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address
LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address

// ----- Adjust the Gamma Curve -----//
LCD_CtrlWrite_ILI9320(0x0030, 0x0007);
LCD_CtrlWrite_ILI9320(0x0031, 0x0504);
LCD_CtrlWrite_ILI9320(0x0032, 0x0703);
LCD_CtrlWrite_ILI9320(0x0035, 0x0002);
LCD_CtrlWrite_ILI9320(0x0036, 0x0707);
LCD_CtrlWrite_ILI9320(0x0037, 0x0406);
LCD_CtrlWrite_ILI9320(0x0038, 0x0006);
LCD_CtrlWrite_ILI9320(0x0039, 0x0404);
LCD_CtrlWrite_ILI9320(0x003C, 0x0700);
LCD_CtrlWrite_ILI9320(0x003D, 0x0A08);
}
```

```
//----- Set GRAM area -----//
LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address

LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // NDL, VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line

//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);

//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);

LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}
```

void LCD_ExitSleep_ILI9320(void)

```
{
//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // R11h=0x0031 at VCI=3.3V DC1[2:0], DC0[2:0], VC[2:0]
    delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013B); // R12h=0x0138 at VCI=3.3V VREG1OUT voltage
    delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1900); // R13h=0x1800 at VCI=3.3V VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x0007); // R29h=0x0008 at VCI=3.3V VCM[4:0] for VCOMH
    delays(50);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}
```

void LCD_EnterSleep_ILI9320(void)

```
{
LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF
//***** Power OFF sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP
}
```

1.3.2. CMO 2.8" initial code

void ILI9320_CMO28_Initial(void)

```
{
    // VCI=2.8V
    //***** Reset LCD Driver *****//
    LCD_nRESET = 1;
        delays(1);          // Delay 1ms
    LCD_nRESET = 0;
        delays(10);         // Delay 10ms    // This delay is necessary
    LCD_nRESET = 1;
        delays(50);         // Delay 50 ms

    //***** Start Initial Sequence *****//

    LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the Vcore voltage and this setting is must.
    LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
    LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
    LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
    LCD_CtrlWrite_ILI9320(0x0003, 0x0030); // set GRAM write direction and BGR=0.
    LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register

    LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
    LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
    LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
    LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
    LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
    LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity

    //*****Power On sequence *****//
    LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0007); // DC1[2:0], DC0[2:0], VC[2:0]
    LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
    LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
        delays(200);          // Dis-charge capacitor power voltage
    LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
        delays(50);          // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0012, 0x013E); // VREG1OUT voltage
        delays(50);          // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0013, 0x1C00); // VDV[4:0] for VCOM amplitude
    LCD_CtrlWrite_ILI9320(0x0029, 0x000C); // VCM[4:0] for VCOMH
        delays(50);
    LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address
    LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address

    // ----- Adjust the Gamma Curve -----//
    LCD_CtrlWrite_ILI9320(0x0030, 0x0007);
    LCD_CtrlWrite_ILI9320(0x0031, 0x0403);
    LCD_CtrlWrite_ILI9320(0x0032, 0x0404);
    LCD_CtrlWrite_ILI9320(0x0035, 0x0002);
    LCD_CtrlWrite_ILI9320(0x0036, 0x0707);
    LCD_CtrlWrite_ILI9320(0x0037, 0x0606);
    LCD_CtrlWrite_ILI9320(0x0038, 0x0106);
    LCD_CtrlWrite_ILI9320(0x0039, 0x0007);
    LCD_CtrlWrite_ILI9320(0x003C, 0x0700);
    LCD_CtrlWrite_ILI9320(0x003D, 0x0707);
}
```

```
//----- Set GRAM area -----//
LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address

LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // NDL, VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line

//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);

//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}
```

void LCD_ExitSleep_ILI9320(void)

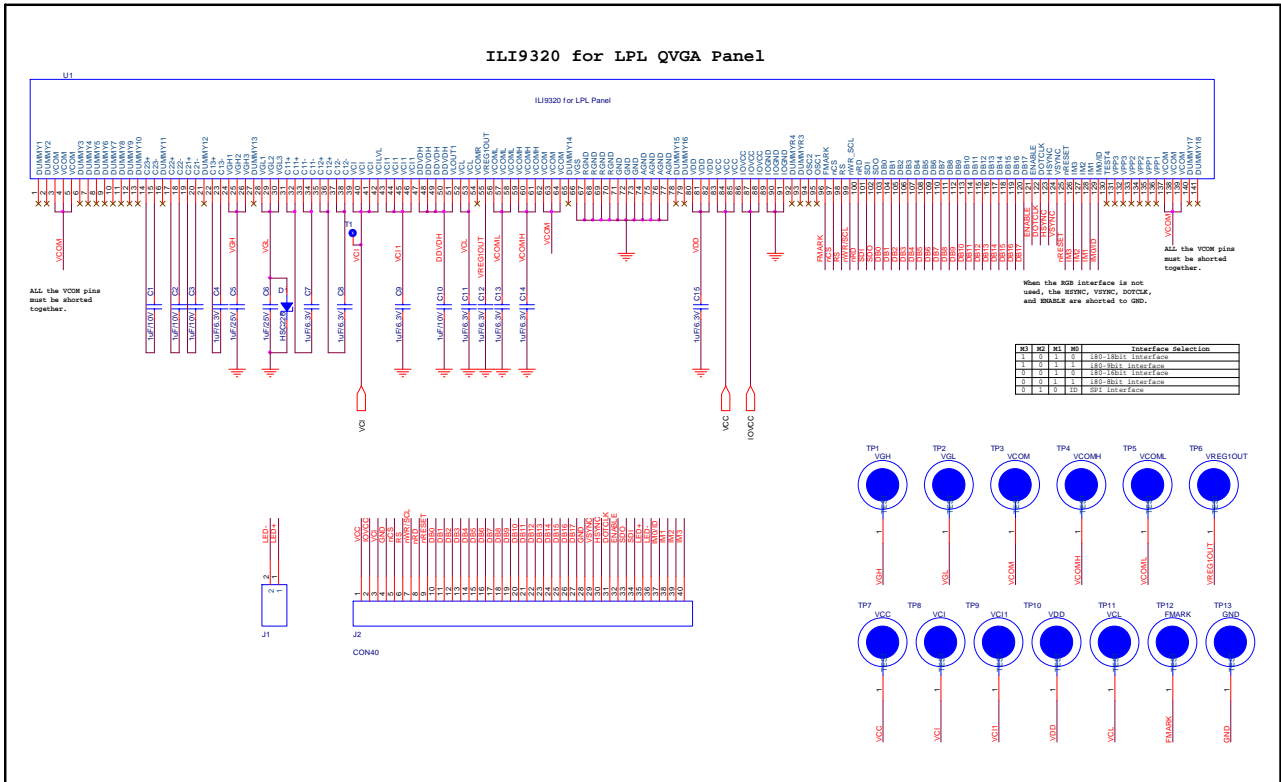
```
{
//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delaysms(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013E); // VREG1OUT voltage
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1E00); // VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000C); // VCM[4:0] for VCOMH
    delaysms(50);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}
```

void LCD_EnterSleep_ILI9320(void)

```
{
LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF

//***** Power OFF sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delaysms(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP
}
```


1.4. LPL2.2" & LPL 2.4" Panel



1.4.1. LPL 2.2" initial code

```

void ILI9320_LPL22_Initial(void)
{
    // VCI=2.8V
    //***** Reset LCD Driver *****//
    LCD_nRESET = 1;
        delaysms(1); // Delay 1ms
    LCD_nRESET = 0;
        delaysms(10); // Delay 10ms
    LCD_nRESET = 1;
        delaysms(50); // Delay 50 ms

    //***** Start Initial Sequence *****//

    LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the internal vcore voltage
    LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
    LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
    LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
    LCD_CtrlWrite_ILI9320(0x0003, 0x1030); // set GRAM write direction and BGR=1.
    LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register

    LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
    LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
    LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
    LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
    LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
    LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity

    //*****Power On sequence *****//
    LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0007); // DC1[2:0], DC0[2:0], VC[2:0]
    LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
    LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
        delaysms(200); // Dis-charge capacitor power voltage

    LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0137); // DC1[2:0], DC0[2:0], VC[2:0]
        delaysms(50); // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0012, 0x013B); // VREG1OUT voltage
        delaysms(50); // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0013, 0x0E00); // VDV[4:0] for VCOM amplitude
    LCD_CtrlWrite_ILI9320(0x0029, 0x000F); // VCM[4:0] for VCOMH
        delaysms(50);

    LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address
    LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address

    // ----- Adjust the Gamma Curve -----//
    LCD_CtrlWrite_ILI9320(0x0030, 0x0307);
    LCD_CtrlWrite_ILI9320(0x0031, 0x0507);
    LCD_CtrlWrite_ILI9320(0x0032, 0x0500);
    LCD_CtrlWrite_ILI9320(0x0035, 0x0304);
    LCD_CtrlWrite_ILI9320(0x0036, 0x0007);
    LCD_CtrlWrite_ILI9320(0x0037, 0x0707);
    LCD_CtrlWrite_ILI9320(0x0038, 0x0007);
    LCD_CtrlWrite_ILI9320(0x0039, 0x0007);
    LCD_CtrlWrite_ILI9320(0x003C, 0x0600);

```

```

LCD_CtrlWrite_ILI9320(0x003D, 0x0700);

//----- Set GRAM area -----//
LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address

LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // ND, VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line

//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);

//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);

LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

void LCD_ExitSleep_ILI9320(void)
{
//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
delayms(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0137); // DC1[2:0], DC0[2:0], VC[2:0]
delayms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013B); // VREG1OUT voltage
delayms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x0E00); // VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000F); // VCM[4:0] for VCOMH
delayms(50);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

void LCD_EnterSleep_ILI9320(void)
{
LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF

//***** Power OFF sequence *****//

```

```
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delayms(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP
```

```
}
```

1.4.2. LPL 2.4" initial code

```

void ILI9320_LPL24_Initial(void)
{
    // VCI=2.8V
    //***** Reset LCD Driver *****//
    LCD_nRESET = 1;
    delayms(1); // Delay 1ms
    LCD_nRESET = 0;
    delayms(10); // Delay 10ms
    LCD_nRESET = 1;
    delayms(50); // Delay 50 ms
    //***** Start Initial Sequence *****//
    LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the internal vcore voltage
    LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
    LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
    LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
    LCD_CtrlWrite_ILI9320(0x0003, 0x1030); // set GRAM write direction and BGR=1.
    LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register
    LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
    LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
    LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
    LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
    LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
    LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity
    //*****Power On sequence *****//
    LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0007); // DC1[2:0], DC0[2:0], VC[2:0]
    LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
    LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delayms(200); // Dis-charge capacitor power voltage
    LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0137); // DC1[2:0], DC0[2:0], VC[2:0]
    delayms(50); // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0012, 0x013C); // VREG1OUT voltage
    delayms(50); // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0013, 0x1600); // VDV[4:0] for VCOM amplitude
    LCD_CtrlWrite_ILI9320(0x0029, 0x0007); // VCM[4:0] for VCOMH
    delayms(50);
    LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address

```

```

LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address
// ----- Adjust the Gamma Curve -----//
LCD_CtrlWrite_ILI9320(0x0030, 0x0007);
LCD_CtrlWrite_ILI9320(0x0031, 0x0403);
LCD_CtrlWrite_ILI9320(0x0032, 0x0400);

LCD_CtrlWrite_ILI9320(0x0035, 0x0003);
LCD_CtrlWrite_ILI9320(0x0036, 0x0007);
LCD_CtrlWrite_ILI9320(0x0037, 0x0606);
LCD_CtrlWrite_ILI9320(0x0038, 0x0106);
LCD_CtrlWrite_ILI9320(0x0039, 0x0007);

LCD_CtrlWrite_ILI9320(0x003C, 0x0700);
LCD_CtrlWrite_ILI9320(0x003D, 0x0700);

//----- Set GRAM area -----//
LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // ND, VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line
//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);
//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

```

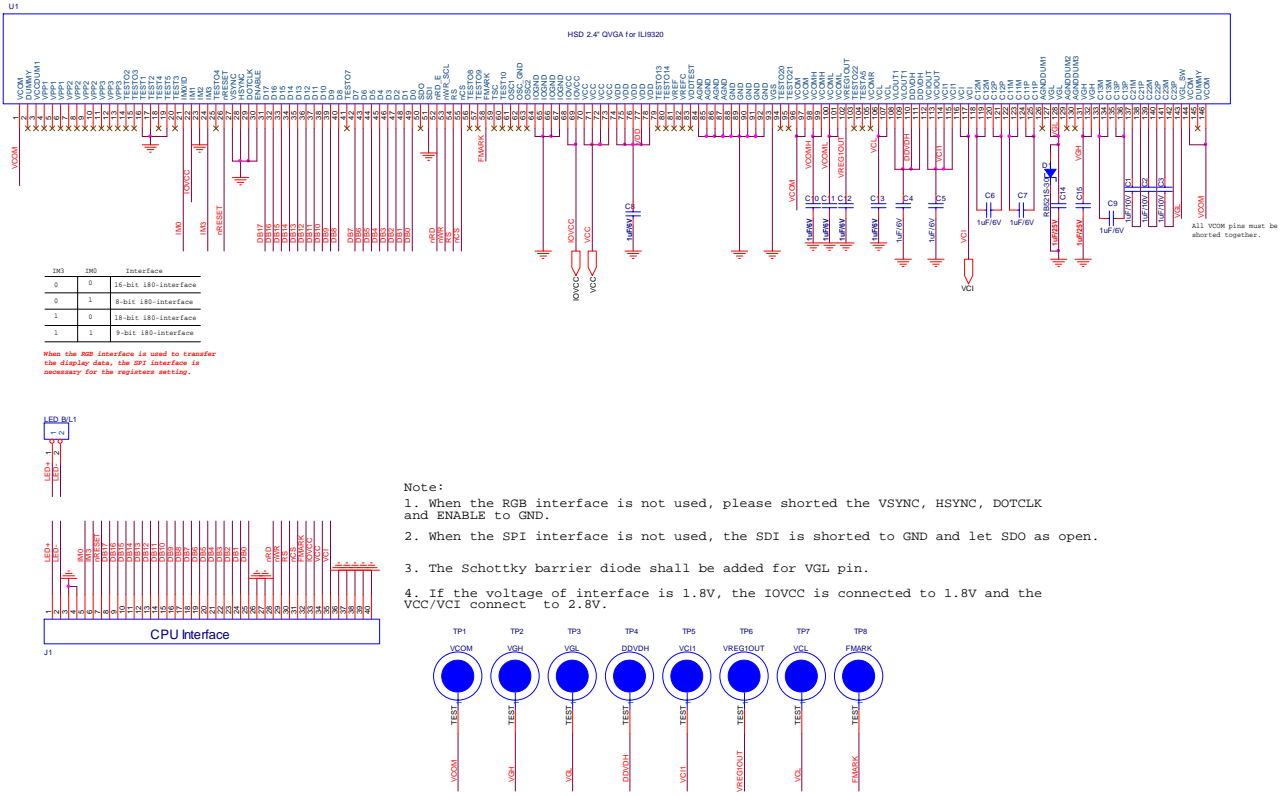
```
void LCD_ExitSleep_ILI9320(void)
{
    //*****Power On sequence *****//
    LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP
    LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
    LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
    LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delayms(200); // Dis-charge capacitor power voltage
    LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
    LCD_CtrlWrite_ILI9320(0x0011, 0x0137); // DC1[2:0], DC0[2:0], VC[2:0]
    delayms(50); // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0012, 0x013C); // VREG1OUT voltage
    delayms(50); // Delay 50ms
    LCD_CtrlWrite_ILI9320(0x0013, 0x1600); //VDV[4:0] for VCOM amplitude
    LCD_CtrlWrite_ILI9320(0x0029, 0x0000); //VCM[4:0] for VCOMH
    delayms(50);
    LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

```

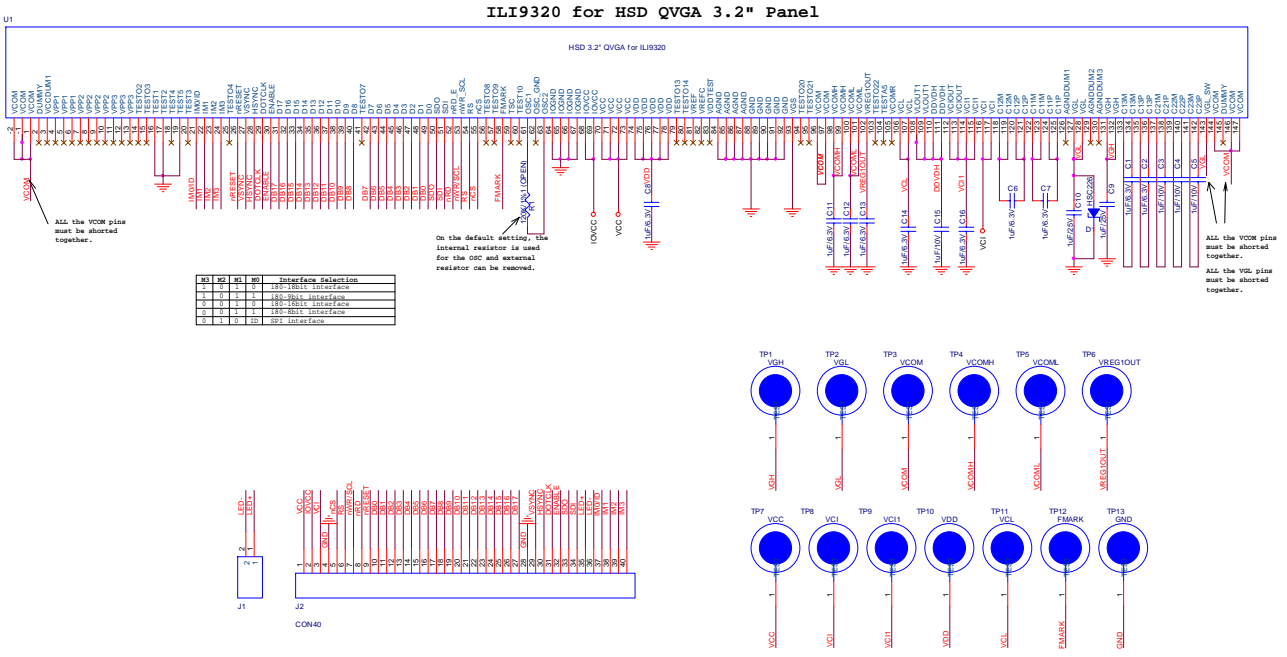
```
void LCD_EnterSleep_ILI9320(void)
{
    LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF
    //***** Power OFF sequence *****//
    LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP
    LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
    LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
    LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delayms(200); // Dis-charge capacitor power voltage
    LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP
}

```

1.5. Hannstar 2.4" Panel



Hannstar 3.2" Panel



HSD 2.4" initial code

```
void ILI9320_HSD24_Initial(void)
```

```
{
// VCI=2.8V
//***** Reset LCD Driver *****//
LCD_nRESET = 1;
    delaysms(1); // Delay 1ms
LCD_nRESET = 0;
    delaysms(10); // Delay 10ms           // This delay time is necessary
LCD_nRESET = 1;
    delaysms(50); // Delay 50 ms

//***** Start Initial Sequence *****//

LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the internal vcore voltage
LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
LCD_CtrlWrite_ILI9320(0x0003, 0x1030); // set GRAM write direction and BGR=1.
LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register

LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity

//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delaysms(200); // Dis-charge capacitor power voltage

LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013A); // VREG1OUT voltage
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1C00); // VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000A); // VCM[4:0] for VCOMH
    delaysms(50);
LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address
LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address

// ----- Adjust the Gamma Curve -----//
LCD_CtrlWrite_ILI9320(0x0030, 0x0007);
LCD_CtrlWrite_ILI9320(0x0031, 0x0203);
LCD_CtrlWrite_ILI9320(0x0032, 0x0001);

LCD_CtrlWrite_ILI9320(0x0035, 0x0007);
LCD_CtrlWrite_ILI9320(0x0036, 0x0407);
LCD_CtrlWrite_ILI9320(0x0037, 0x0607);
LCD_CtrlWrite_ILI9320(0x0038, 0x0106);
LCD_CtrlWrite_ILI9320(0x0039, 0x0007);
}
```

```

LCD_CtrlWrite_ILI9320(0x003C, 0x0007);
LCD_CtrlWrite_ILI9320(0x003D, 0x001E);

//----- Set GRAM area -----//
LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address

LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // NDL, VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line

//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);

//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);

LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

void LCD_ExitSleep_ILI9320(void)
{
//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // R11h=0x0031 at VCI=3.3V DC1[2:0], DC0[2:0], VC[2:0]
delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013A); // R12h=0x0138 at VCI=3.3V VREG1OUT voltage
delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1C00); // R13h=0x1800 at VCI=3.3V VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000A); // R29h=0x0008 at VCI=3.3V VCM[4:0] for VCOMH
delays(50);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

void LCD_EnterSleep_ILI9320(void)
{
LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF
//***** Power OFF sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP

```

```

LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP
}

```

HSD 3.2" initial code

void ILI9320_HSD32_Initial(void)

```

{
// VCI=2.8V
//***** Reset LCD Driver *****//
LCD_nRESET = 1;
delays(1); // Delay 1ms
LCD_nRESET = 0;
delays(10); // Delay 10ms // This delay time is necessary
LCD_nRESET = 1;
delays(50); // Delay 50 ms

//***** Start Initial Sequence *****//

LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the internal vcore voltage
LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
LCD_CtrlWrite_ILI9320(0x0003, 0x1030); // set GRAM write direction and BGR=1.
LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register

LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity

//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
delays(200); // Dis-charge capacitor power voltage

LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013B); // VREG1OUT voltage
delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1800); // VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000F); // VCM[4:0] for VCOMH
delays(50);
LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address
LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address

// ----- Adjust the Gamma Curve -----//
LCD_CtrlWrite_ILI9320(0x0030, 0x0000);

```

```

LCD_CtrlWrite_ILI9320(0x0031, 0x0007);
LCD_CtrlWrite_ILI9320(0x0032, 0x0103);

LCD_CtrlWrite_ILI9320(0x0035, 0x0407);
LCD_CtrlWrite_ILI9320(0x0036, 0x090F);
LCD_CtrlWrite_ILI9320(0x0037, 0x0404);
LCD_CtrlWrite_ILI9320(0x0038, 0x0400);
LCD_CtrlWrite_ILI9320(0x0039, 0x0404);

LCD_CtrlWrite_ILI9320(0x003C, 0x0000);
LCD_CtrlWrite_ILI9320(0x003D, 0x0400);

//----- Set GRAM area -----//
LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address

LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // NDL,VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line

//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);

//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);

LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}

```

void LCD_ExitSleep_ILI9320(void)

```

{
//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
    delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x013B); // VREG1OUT voltage
    delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1800); // VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000F); // VCM[4:0] for VCOMH
    delays(50);
}

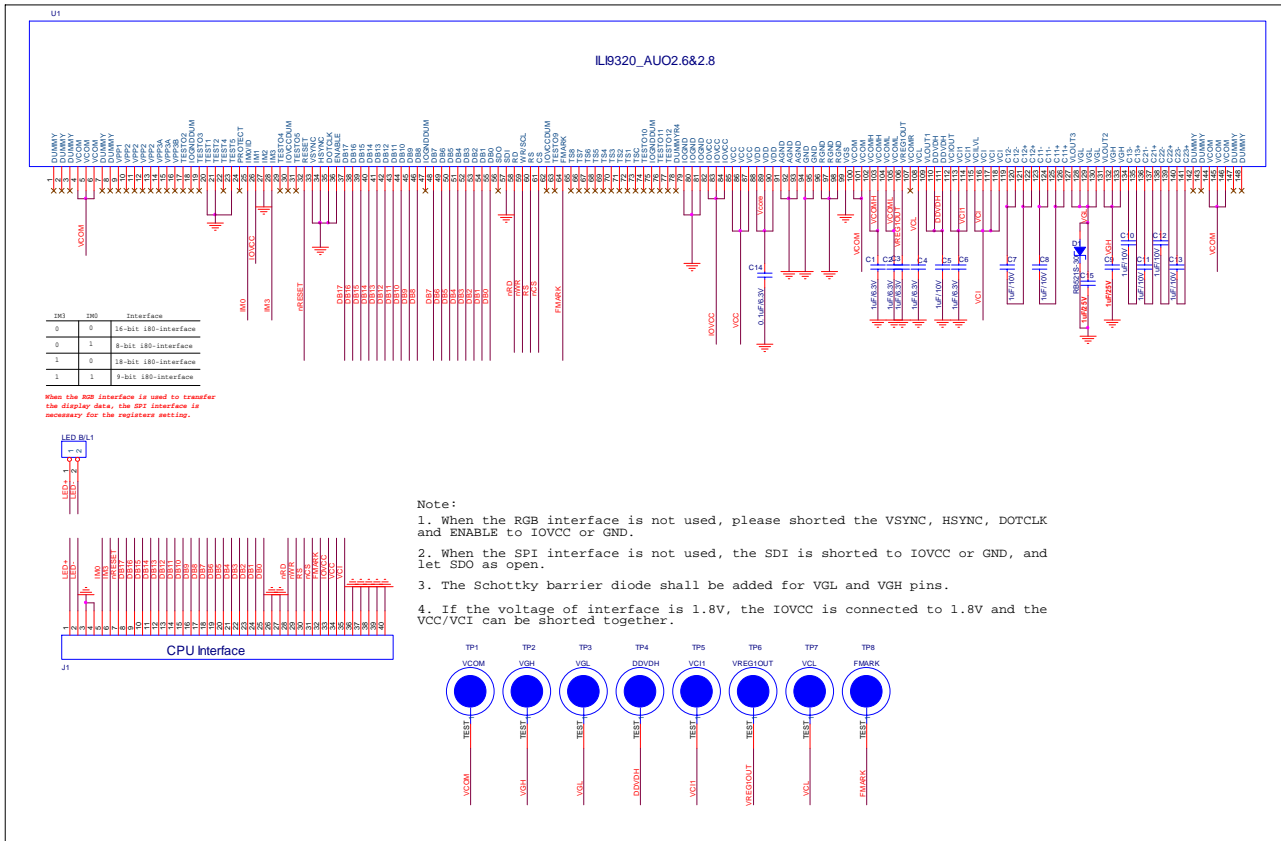
```

```
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON  
}
```

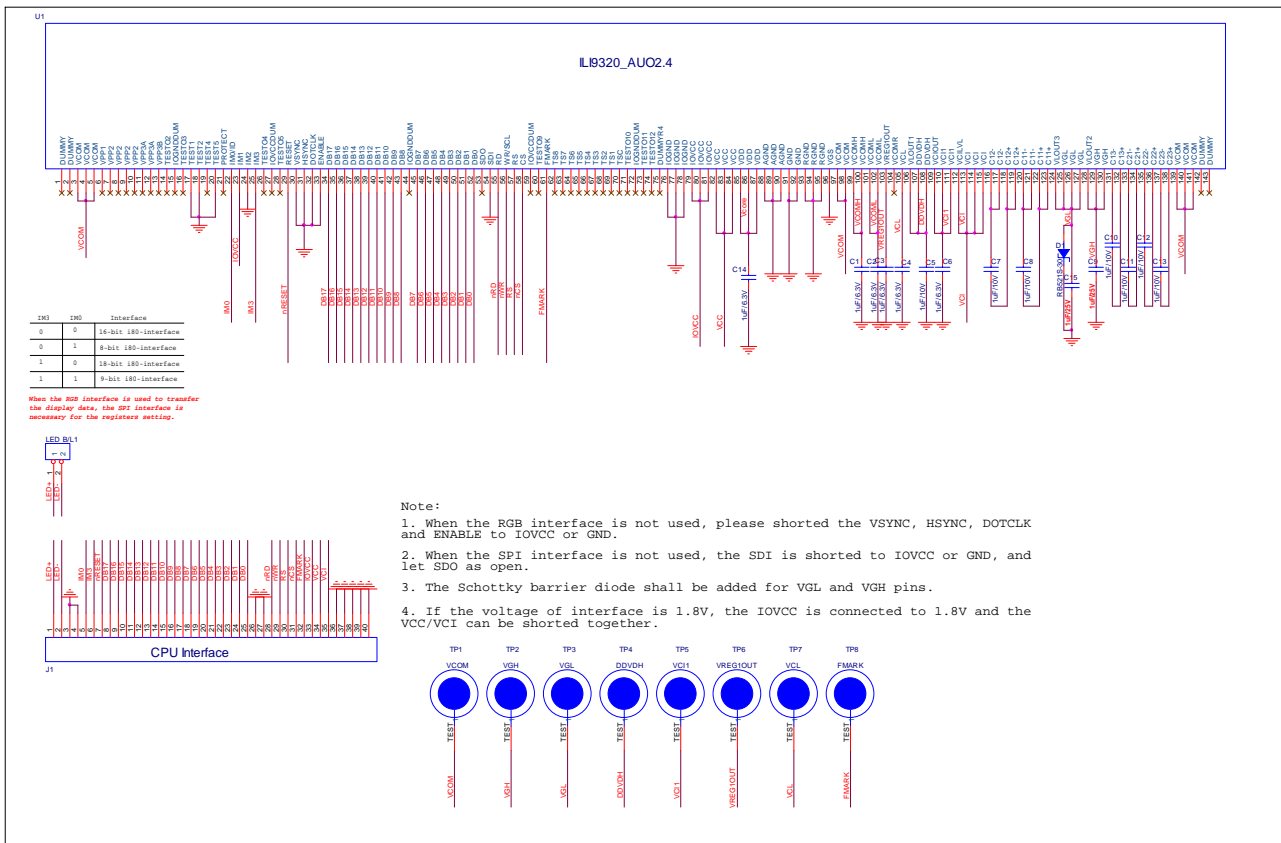
void LCD_EnterSleep_ILI9320(void)

```
{  
LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF  
//***** Power OFF sequence *****//  
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP  
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]  
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage  
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude  
    delays(200); // Dis-charge capacitor power voltage  
LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP  
}
```

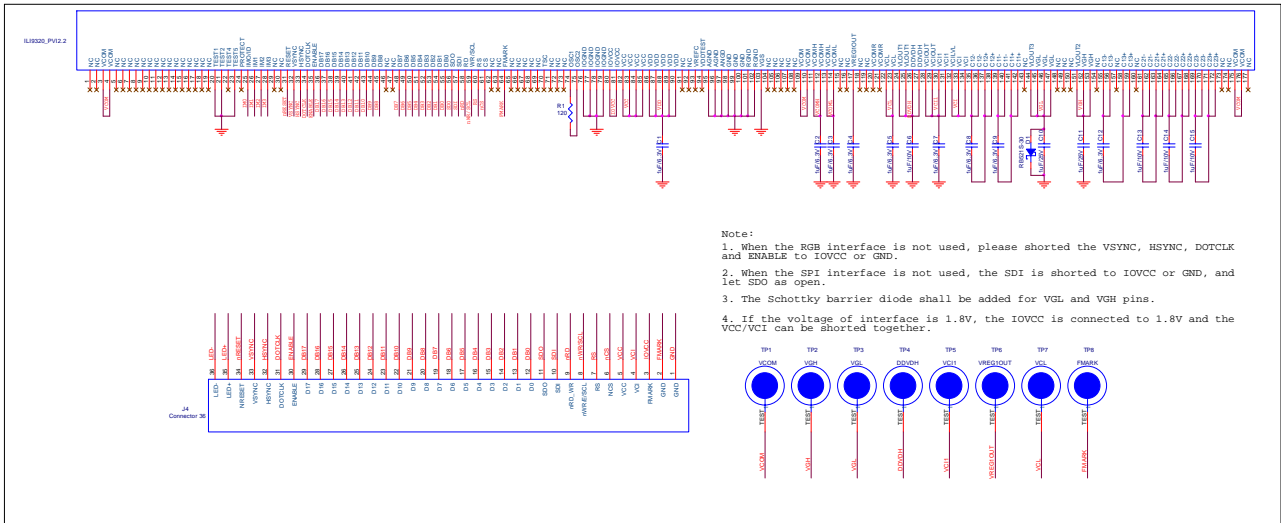
1.6. AUO 2.6" & 2.8" Panel



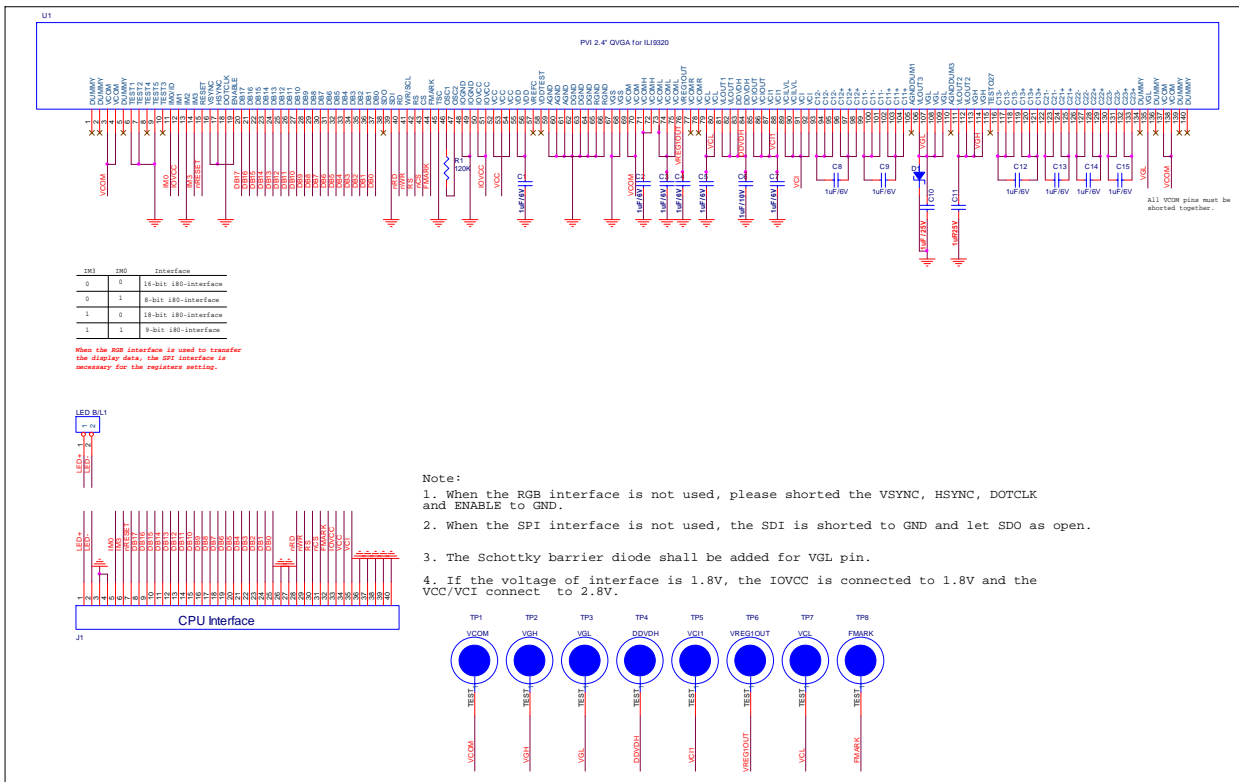
AUO 2.4" Panel



1.7. PVI2.2" 2.8" Panel



PVI2.4" Panel



1.7.1 PVI 2.8" initial code

void ILI9320_PVI28_Initial(void)

```
{
// VCI=2.8V
//***** Reset LCD Driver *****//
LCD_nRESET = 1;
    delaysms(1); // Delay 1ms
LCD_nRESET = 0;
    delaysms(10); // Delay 10ms           // This delay time is necessary
LCD_nRESET = 1;
    delaysms(50); // Delay 50 ms
//***** Start Initial Sequence *****//

LCD_CtrlWrite_ILI9320(0x00E5, 0x8000); // Set the internal vcore voltage
LCD_CtrlWrite_ILI9320(0x0000, 0x0001); // Start internal OSC.
LCD_CtrlWrite_ILI9320(0x0001, 0x0100); // set SS and SM bit
LCD_CtrlWrite_ILI9320(0x0002, 0x0700); // set 1 line inversion
LCD_CtrlWrite_ILI9320(0x0003, 0x1030); // set GRAM write direction and BGR=1.
LCD_CtrlWrite_ILI9320(0x0004, 0x0000); // Resize register

LCD_CtrlWrite_ILI9320(0x0008, 0x0202); // set the back porch and front porch
LCD_CtrlWrite_ILI9320(0x0009, 0x0000); // set non-display area refresh cycle ISC[3:0]
LCD_CtrlWrite_ILI9320(0x000A, 0x0000); // FMARK function
LCD_CtrlWrite_ILI9320(0x000C, 0x0000); // RGB interface setting
LCD_CtrlWrite_ILI9320(0x000D, 0x0000); // Frame marker Position
LCD_CtrlWrite_ILI9320(0x000F, 0x0000); // RGB interface polarity

//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delaysms(200); // Dis-charge capacitor power voltage

LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // DC1[2:0], DC0[2:0], VC[2:0]
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x0138); // VREG1OUT voltage
    delaysms(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1700); // VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000D); // VCM[4:0] for VCOMH
    delaysms(50);
LCD_CtrlWrite_ILI9320(0x0020, 0x0000); // GRAM horizontal Address
LCD_CtrlWrite_ILI9320(0x0021, 0x0000); // GRAM Vertical Address

// ----- Adjust the Gamma Curve -----//
LCD_CtrlWrite_ILI9320(0x0030, 0x0001);
LCD_CtrlWrite_ILI9320(0x0031, 0x0606);
LCD_CtrlWrite_ILI9320(0x0032, 0x0304);
LCD_CtrlWrite_ILI9320(0x0035, 0x0103);
LCD_CtrlWrite_ILI9320(0x0036, 0x011D);
LCD_CtrlWrite_ILI9320(0x0037, 0x0404);
LCD_CtrlWrite_ILI9320(0x0038, 0x0404);
LCD_CtrlWrite_ILI9320(0x0039, 0x0404);
LCD_CtrlWrite_ILI9320(0x003C, 0x0700);
LCD_CtrlWrite_ILI9320(0x003D, 0x0A1F);
}
```



```
//----- Set GRAM area -----//
LCD_CtrlWrite_ILI9320(0x0050, 0x0000); // Horizontal GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0051, 0x00EF); // Horizontal GRAM End Address
LCD_CtrlWrite_ILI9320(0x0052, 0x0000); // Vertical GRAM Start Address
LCD_CtrlWrite_ILI9320(0x0053, 0x013F); // Vertical GRAM Start Address

LCD_CtrlWrite_ILI9320(0x0060, 0x2700); // Gate Scan Line
LCD_CtrlWrite_ILI9320(0x0061, 0x0001); // NDL, VLE, REV
LCD_CtrlWrite_ILI9320(0x006A, 0x0000); // set scrolling line

//----- Partial Display Control -----//
LCD_CtrlWrite_ILI9320(0x0080, 0x0000);
LCD_CtrlWrite_ILI9320(0x0081, 0x0000);
LCD_CtrlWrite_ILI9320(0x0082, 0x0000);
LCD_CtrlWrite_ILI9320(0x0083, 0x0000);
LCD_CtrlWrite_ILI9320(0x0084, 0x0000);
LCD_CtrlWrite_ILI9320(0x0085, 0x0000);

//----- Panel Control -----//
LCD_CtrlWrite_ILI9320(0x0090, 0x0010);
LCD_CtrlWrite_ILI9320(0x0092, 0x0000);
LCD_CtrlWrite_ILI9320(0x0093, 0x0003);
LCD_CtrlWrite_ILI9320(0x0095, 0x0110);
LCD_CtrlWrite_ILI9320(0x0097, 0x0000);
LCD_CtrlWrite_ILI9320(0x0098, 0x0000);

LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}
```

void LCD_ExitSleep_ILI9320(void)

```
{
//*****Power On sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x17B0); // SAP, BT[3:0], AP, DSTB, SLP, STB
LCD_CtrlWrite_ILI9320(0x0011, 0x0037); // R11h=0x0031 at VCI=3.3V DC1[2:0], DC0[2:0], VC[2:0]
    delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0012, 0x0138); // R12h=0x0138 at VCI=3.3V VREG1OUT voltage
    delays(50); // Delay 50ms
LCD_CtrlWrite_ILI9320(0x0013, 0x1700); // R13h=0x1800 at VCI=3.3V VDV[4:0] for VCOM amplitude
LCD_CtrlWrite_ILI9320(0x0029, 0x000D); // R29h=0x0008 at VCI=3.3V VCM[4:0] for VCOMH
    delays(50);
LCD_CtrlWrite_ILI9320(0x0007, 0x0173); // 262K color and display ON
}
```

void LCD_EnterSleep_ILI9320(void)

```
{
LCD_CtrlWrite_ILI9320(0x0007, 0x0000); // display OFF
//***** Power OFF sequence *****//
LCD_CtrlWrite_ILI9320(0x0010, 0x0000); // SAP, BT[3:0], APE, AP, DSTB, SLP
LCD_CtrlWrite_ILI9320(0x0011, 0x0000); // DC1[2:0], DC0[2:0], VC[2:0]
LCD_CtrlWrite_ILI9320(0x0012, 0x0000); // VREG1OUT voltage
LCD_CtrlWrite_ILI9320(0x0013, 0x0000); // VDV[4:0] for VCOM amplitude
    delays(200); // Dis-charge capacitor power voltage
LCD_CtrlWrite_ILI9320(0x0010, 0x0002); // SAP, BT[3:0], APE, AP, DSTB, SLP
}
```

Revision History

Revision History

Version No.	Date	Page	Description
V0.1	2006/11/22		New
V0.2	2006/11/28		Add the Hannstar FPC circuit
V0.4	2007/01/02		Add CPT 2.8" initial code
V0.5	2007/1/5		Modify the Hannstar FPC circuit
V0.6	2007/2/2		Add the Hannstar initial code
V0.7	2007/2/7		Add the AUO FPC circuit
V0.8	2007/2/9		ADD PVI FPC circuit
V0.9	2007/3/13		ADD CPT 3.2" panel FPC circuit and initial code
V0.91	2007/03/20		ADD Hannstar 2.4 initial code
V0.92	2007/04/21		ADD Hanstar 3.2 FPC and Initial code
			ADD PVI 2.4 FPC and PVI 2.8 Initial code
			ADD CMO 2.2 and 2.4 Initial code