

## 1、概述

FYD12864-0402 采用 ST7920 控制芯片，并提供了中文字库，为中文显示开发方面带来了更多的方便。现以 FYD12864-0402 为例说明 FYD12864-0402 系列液晶显示模块的应用。

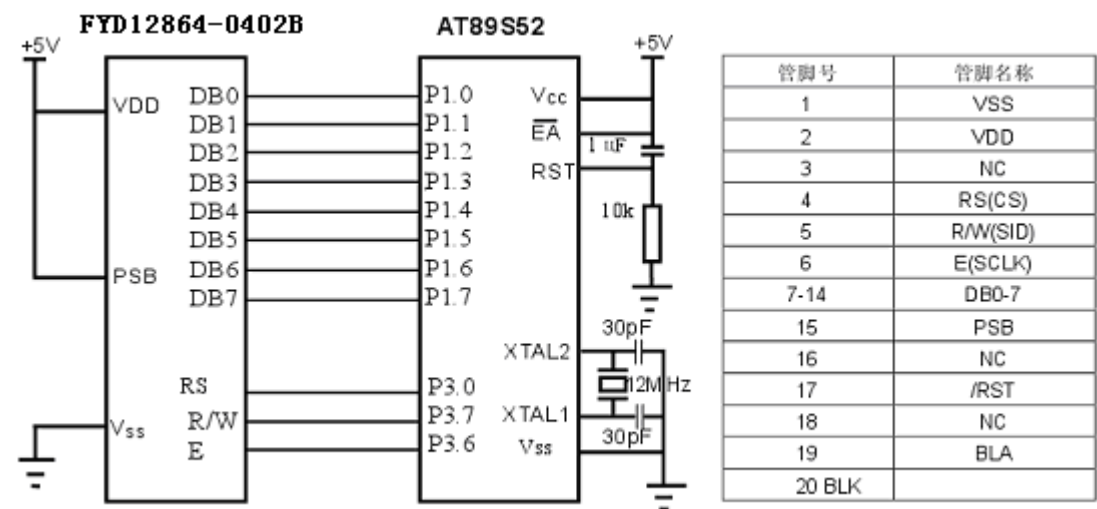
该模块的主要特性有：

- 汉字显示：内置汉字字库、提供 8192 个 16×16 点阵汉字(简体)
- 半宽字型显示：内置 128 个 16×8 点阵字符
- 绘图显示：绘图显示画面提供一个 64×256 点的绘图区域 GDRAM
- 自定义字型显示：含 CGRAM 提供 2 组软件可编程的 16×16 点阵造字功能
- 电源电压：5V 单电源供电
- 显示分辨率：128×64 点
- 显示方式：STN、正显、半透
- 显示颜色：白底蓝字
- 驱动方式：1/33DUTY、1/5BIAS
- 通讯方式：8/4 位并行方式或串行方式
- 工作温度：-20℃ - +70℃、存储温度：-30℃ - +80℃

该模块可满足系统各种工作电压及便携式仪器低功耗的要求。广泛用于各种仪器仪表、家用电器和信息产品上作为显示器件。

## 2、硬件接口

FYD12864-0402 与单片机的联接我们采用的是 8 位并行接口，此时“PSB”脚必须接高电平。在并行模式下可由功能设定指令中的“DL”位来选择 8 位或 4 接口方式（DL=1 为 8 位接口），主控制系统将配合“RS”、“RW”、“E” DB0～DB7 来完成指令/数据的传送，其操作时序与其它并行接口液晶显示模块相同。



注：1.模块与单片机连结如果为串行“PSB”脚必须接低电平

2.如果液晶显示模块采取 3V 供电则模块引脚的脚 3 和脚 18 之间必须加一可调电阻

### 3、控制器接口信号说明

RS, R/W 和 E 信号的配合选择决定控制界面的 4 种模式:

RS	R/W	E	DB0-DB7	功能说明
L	L	H→L	输入态	MPU 写指令到指令寄存器 (IR)
L	H	H	输出态	读出忙标志 (BF) 及地址计数器 (AC) 的状态
H	L	H→L	输入态	MPU 写入数据到数据寄存器 (DR)
H	H	H	输出态	MPU 从数据寄存器 (DR) 中读出数据

### 4、应用说明

用 FYD12864-0402 显示模块时应注意以下几点:

- ① 用户指令集. 时序图. 软件的初始化: 详见 LCD 控制/驱动器 ST7920
- ② 欲在某一个位置显示中文字符时, 应先设定显示字符位置, 即先设定显示地址, 再写入中文字符编码。
- ③ 显示 ASCII 字符过程与显示中文字符过程相同, 不过在显示连续字符时, 只须设定一次显示地址, 由模块自动对地址加 1 指向下一个字符位置, 否则, 显示的字符中将会有一个空 ASCII 字符位置。
- ④ 当字符编码为 2 字节时, 应先写入高位字节, 再写入低位字节。
- ⑤ “RE” 为基本指令集与扩充指令集的选择控制位。当变更 “RE” 后, 以后 的指令集将维持在最后的状态, 除非再次变更 “RE” 位, 否则使用相同指令集时, 无需每次均重设 “RE” 位。

### 5、应用举例

#### 5.1 字符显示

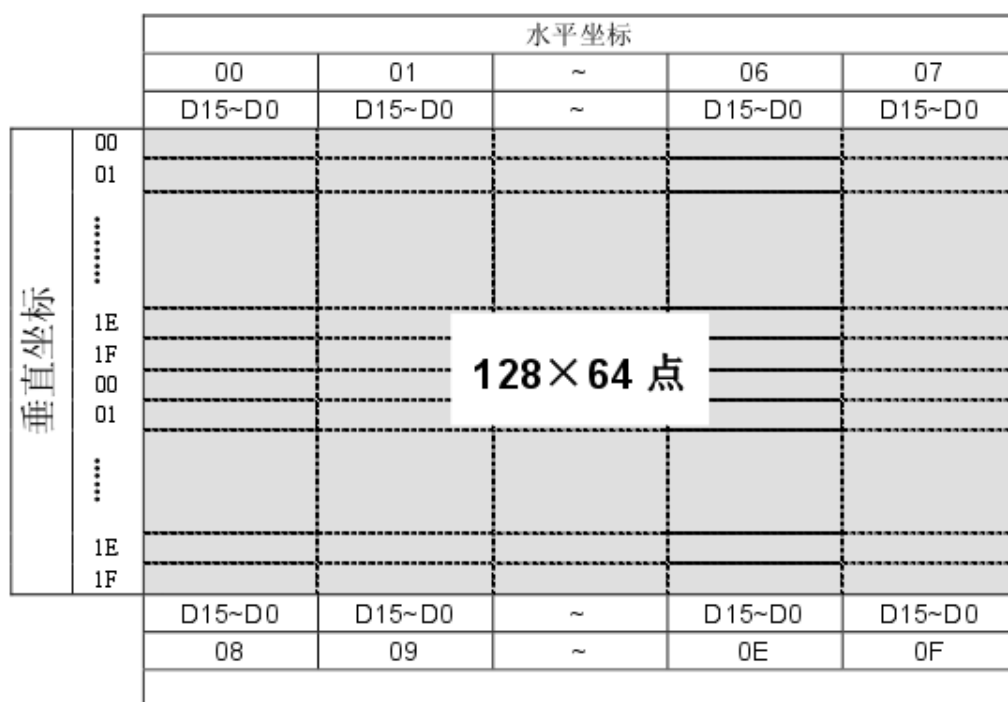
每屏可显示 4 行 8 列共 32 个 16×16 点阵的汉字, 每个显示 RAM 可显示 1 个中文字符或 2 个 16×8 点阵全高 ASCII 码字符, 即每屏最多可实现 32 个中文字符或 64 个 ASCII 码字符的显示。内部提供 128×2 字节的字符显示 RAM 缓冲区 (DDRAM)。字符显示是通过将字符显示编码写入该字符显示 RAM 实现的。根据写入内容的不同, 可分别在液晶屏上显示 CGROM (中文字库)、HCGROM (ASCII 码字库) 及 CGRAM (自定义字形) 的内容。三种不同字符/字型的选择编码范围为: 显示自定义字型其代码分别是 0000H、0002H、0004H 和 0006H 共 4 个, 显示半宽 ASCII 码字符为 02H~7FH, A1A0H~F7FFH 显示 8192 种 GB2312 中文字库字形。字符显示 RAM 在液晶模块中的地址 80H~9FH。字符显示的 RAM 的地址与 32 个字符显示区域有着一一对应的关系, 其对应关系如下表所示。

80H	81H	82H	83H	84H	85H	86H	87H
90H	91H	92H	93H	94H	95H	96H	97H
88H	89H	8AH	8BH	8CH	8DH	8EH	8FH
98H	99H	9AH	9BH	9CH	9DH	9EH	9FH

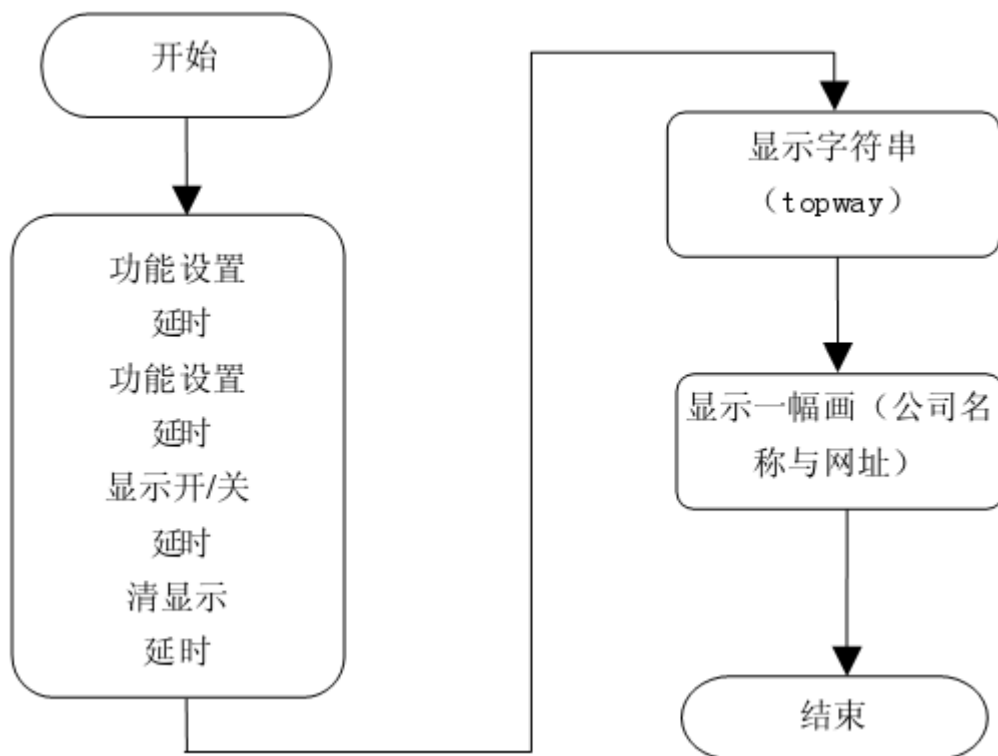
## 5.2 图形显示

绘图显示 RAM 提供 64x32 个位元组的记忆空间(由扩充指令设定绘图 RAM 位址),在更改绘图 RAM 时,由扩充指令设定 GDRAM 位址先设置垂直位址,再设置水平位址(连续写入两个位元组的资料来完成垂直与水平的坐标位址),再写入两个 8 位的资料到绘图 RAM,而位址计数器(AC)会自动加一,整个写入绘图 RAM 的步骤如下:

- 先将垂直的位元组(Y)写入绘图 RAM 位址
- 再将水平的位元组(X)写入绘图 RAM 位址.
- 将 D15~D8 写入到 RAM 中
- 将 D7~D0 写入到 RAM 中



5.3 参考程序设计：软件实现 LCD 的显示，显示字符，图象。程序主流程图如下：



#### 5.4 参考程序

```

#include <reg.52>
#include <intrins.h>
sbit RS = P3^0;
sbit RW = P3^7;
sbit E = P3^6;
#define Lcd_Bus P1 //MCU P1<-----> LCM
//延时子程序
void delay(unsigned int t)
{ unsigned int i,j;
  for(i=0;i
  for(j=0;j
)

//写命令到 LCD
void write_com(unsigned char cmdcode)
{ RS=0;

  RW=0;
  E=1;
  Lcd_Bus=cmdcode;
  E=0;
  delay(5);

```

```

    }

    //写数据到 LCD
    void write_data(unsigned char Dispdata)
    { RS=1;

    RW=0;
    E=1;
    Lcd_Bus=Dispdata;
    E=0;
    delay(5);
    //初始化 LCD 屏
    void lcdinit()
    { delay(2000);
    write_com(0x30);
    delay(10); //选择基本指令集
    write_com(0x30); //选择 8bit 数据流
    delay(5);
    write_com(0x0c); //开显示(无游标、不反白)
    delay(10);
    write_com(0x01); //清显示并设地址指针为 00H
    delay(500);
    write_com(0x06); //指定在资料的读取及写入时

    //设定游标的移动方向及指
    //定显示的移位
    delay(0);
    }

    //显示字符串
    void hzkdis(unsigned char code *s)
    { while(*s>0)
    { write_data(*s);
    s++;

    delay(50);
    }
    }

    //显示文本（FYD12864-0402B 显示程序）
    void ceshi()
    { write_com(0x01);
    delay(5);
    write_com(0x82);

```

```

hzkdis("CDFYD");
write_com(0x88);
hzkdis("FYD12864 测试程序");
}

//显示图片
void Disp_Img(unsigned char code *img)
{ unsigned int j=0;
  unsigned char x,y,i;
  for(i=0;i
  for(y=0;y
  for(x=0;x
  { write_com(0x36);
    write_com(y+0x80);
    write_com(x+0x80+i);

    write_com(0x30);
    write_data(img[j++]);
    write_data(img[j++]);}
  }

  unsigned char code BMP[]={/*显示数据略*/};

  main()
  { lcdinit (); //初始化 LCD 屏
    delay(10);
    while(1)
    {
      ceshi(); //显示文本
      delay(5000);
      write_com(0x01); //清屏
      delay(10);
      Disp_Img(BMP); //调入图画
      delay(10000);
      write_com(0x01); //清屏
      delay(10);
    }
  } //End Of Program

```