程序一大，应用一广，就会出现许多问题，例如以前花很多功夫写的示波器SPI LCD屏程序，现在完全不能用在遥控器的ILI9341 LCD屏上，即使要用，都要修改许多库（字符显示、画圆、区域填充、画波形曲线…）并且都要测试通过。这就让许多工作都是白费的，自然想到提高通用性，那么程序分层自然就来了：

字符显示函数

void ShowChar()

{不涉及IO或特定显示屏寄存器的操作；而是使用更下一层库提供的函数接口，比如画一个点DrawPoint（）；}

画点函数

void DrawPoint（）

{

具体显示屏的IO操作；

}

ILI9341 LCD

画点函数

void DrawPoint（）

{

具体显示屏的IO操作；

}

另一种SPI LCD

…

LCD驱动库

…

…

显示库

**另一个包含LCD的设计中还可以用ShowChar**

当然LCD的IO操作程序等都是写到主控芯片中的，需要包含主控芯片的特色IO操作，比如KL26对CS引脚的操作写为：PTB1\_OUT = X, STM32则是PD5\_OUT = X;当更换了芯片后也只需切换LCD 库包含的头文件即可，工作量已经小很多了，而且以前的工作终于可以沿用下去了！

## 宏定义

宏需要考虑的一点是，该参数在程序种是否一成不变

## 函数所在层

如果不确定函数应该写在哪一层，想想换个LCD是否需要该函数

## 通专用性