

LVGL移植

LVGL移植关键点

- 显示设备 (DMA2D)
- 输入设备
- 操作系统
- 内存
 - 外部SRAM
 - 内存管理算法

LVGL移植整体流程

- 1、确定输入、输出设备
 - 输出设备：显示器
 - 输入设备：键盘、鼠标、触摸屏、按键以及编码器
- 2、确定所需功能
 - 屏幕数据传输方式
 - 系统
 - SRAM
 - 内存管理算法
- 3、准备LVGL库、例程
 - V8.2版本的LVGL库文件
 - 支持所需功能的例程源码
- 4、按需裁剪、修改LVGL库文件，并添加到MDK工程中
- 5、适配自己的输入和输出设备，添加所需功能（例如系统）
- 6、为LVGL提供时基，写测试代码检测移植是否成功

LVGL移植演示

- 课堂代码1
 - 功能：带触摸、裸机、内部SRAM、LVGL自带内存管理
 - 移植时需要注意的点：
 - 1、移植时新建LVGL工程文件夹的结构
 - 2、必须开启C99模式
 - 3、在定时器中断里调用：lv_tick_inc(x)，需保证：进入中断的时间间隔 = x 毫秒
 - 4、如果使用官方的demo，需要在lv_conf.h中开启
- 课堂代码2
 - 功能：带触摸、FreeRTOS、内部SRAM、LVGL自带内存管理
 - 移植时需要注意的点：
 - 1、port.c文件需要根据具体芯片进行选择
 - 2、需要在lv_conf.h文件中配置自定义的时钟源
- 课堂代码3
 - 功能：带触摸、FreeRTOS、外部SRAM、自研内存管理算法
 - 移植时需要注意的点：
 - 1、不同的开发板外扩的内存类型不同，有SRAM和SDRAM
 - 2、非常不推荐把LVGL管理的内存空间放到外部SRAM
 - 3、分配给LVGL管理的内存不能太多或者太少，需要兼顾图形缓冲区的内存
- 课堂代码4
 - 功能：带触摸、FreeRTOS、DMA2D、内部SRAM、LVGL自带内存管理
 - 移植时需要注意的点：
 - 1、如果使用的是MCU的屏幕，一定不要开启DMA2D
 - 2、需要更直观地观察优化效果，可以开启帧率显示
 - 3、选择更高等级的优化和开启优化时间选项，LVGL运行效果会更好

LVGL移植相关文件介绍 (lv_conf.h)

- 1、颜色板块：颜色深度、字节交换、屏幕透明等
- 2、内存板块：内存管理算法选择、内存分配大小等
- 3、硬件层板块：显示刷新周期、输入设备读取周期等
- 4、特征板块：绘图、日志、帧率显示等
- 5、编译器板块：大数组前缀、内存对齐等
- 6、字体板块：开启系统字体、配置自定义字体等
- 7、文本板块：字符编码、文本特性
- 8、核心部件板块：使能/失能核心部件
- 9、拓展功能：使能/失能拓展部件、第三方库等
- 10、实例：使能/失能LVGL官方的实例