

正点原子 littleVGL 开发指南

lv_ddlist 下拉列表框

开发指南

正点原子
广州市星翼电子科技有限公司

修订历史

| 版本 | 日期 | 原因 |
|-------|------------|-------|
| V1.00 | 2020/05/01 | 第一次发布 |

lv_ddlist 下拉列表框

1. 介绍

lv_ddlist 是一个下拉列表框控件,它允许用户从多个选项值中去任意选择一个值,它具有收缩和展开俩个状态,默认情况下,它是处于收缩状态的,在收缩状态下,它会将当前选择的值给显示出来,当通过 API 接口或者通过用户点击使其处于展开状态时,它会将所有的选项值都给显示出来,然后用户可以从中选择一个想要的值,一旦选择好之后, lv_ddlist 下拉列表框控件会自动从展开状态回到收缩状态的.如果你想通过 API 接口来手动改变下拉列表框控件的状态,那也是可以的,这是通过 lv_ddlist_open(ddlist)和 lv_ddlist_close(ddlist)接口来完成的,收缩和展开这俩种状态的外观图如下所示:



图 1.1 下拉列表框控件的收缩状态



图 1.2 下拉列表框控件的展开状态

你可以通过 lv_ddlist_set_options(ddlist, options)接口来给下拉列表框控件设置所有的选项值,其中 options 是一个字符串,而不是一个数组,所以每一个选项值之间需要用'\n'换行符来作为区分,如 const char * options = "First\nSecond\nThird";就定义了 First, Secode, Third 这三个选项值,除了通过用户点击来选择选项值之外,我们还可以直接通过 lv_ddlist_set_selected(ddlist, id)接口来选择某一个选项值,相反的,你可以通过 lv_ddlist_get_selected(ddlist)接口来获取下拉列表框控件当前选择的选项值,这个 API 接口返回的是选项值在下拉列表框中的索引位置,如果你想获取它的文本内容的话,那么你可以通过 lv_ddlist_get_selected_str(ddlist, buf, buf_size)接口来完成.

在图 1.1 中所示的向下箭头默认情况下是没有的,它需要通过调用 lv_ddlist_set_draw_arrow(ddlist, true)接口来使能绘制向下箭头,然后下拉列表框中文本的水平对齐方式是通过 lv_ddlist_set_align(ddlist, LV_LABEL_ALIGN_LEFT/CENTER/RIGHT)接口来设置的,然后这里需要来讲一下如何设置下拉列表框控件的大小,大家的第一印象可能会想到使用 lv_obj_set_size 接口,但是我告诉大家,这是无效的,因为下拉列表框控件提供了专门的 API 接口来设置其大小,设置宽度的话用 lv_ddlist_set_fix_width(ddlist, width)接口,设置高度的话用 lv_ddlist_set_fix_height(ddlist, height)接口,如果不设置宽和高的话,也是没有关系的,因为下拉列表框控件会自动根据选项值的多少和文本内容的长短来自动确定其高和宽的.当你设置的高度不足以展示所有的选项值时,下拉列表框控件就会出现滚动条,你可以通过 lv_ddlist_set_sb_mode(ddlist, LV_SB_MODE_...)接口来设置滚动条的模式.

最后我们来说一下下拉列表框控件的事件,当它的选项值被改变时,它会给它的事件回调函数发送一个 LV_EVENT_VALUE_CHANGED 事件.

2. lv_ddlist 的 API 接口

2.1 主要数据类型

2.1.1 下拉列表框样式数据类型

```
enum {  
    LV_DDLIST_STYLE_BG,  
    LV_DDLIST_STYLE_SEL,  
    LV_DDLIST_STYLE_SB,  
};  
typedef uint8_t lv_ddlist_style_t;
```

LV_DDLIST_STYLE_BG: 用来修饰下拉列表框控件的背景,使用样式中的 body 和 text 字段,默认值为 lv_style_pretty

LV_DDLIST_STYLE_SEL: 用来修饰被选择的选项值,使用样式中的 body 和 text 字段,默认值为 lv_style_plain_color

LV_DDLIST_STYLE_SB: 用来修饰滚动条的,默认值为 lv_style_plain_color,具体用法请参考 lv_page 页面章节

2.2 API 接口

2.2.1 创建对象

```
lv_obj_t * lv_ddlist_create(lv_obj_t * par, const lv_obj_t * copy);
```

参数:

par: 父对象

copy: 拷贝的对象,如果无拷贝的话,传 NULL 值

返回值:

返回创建出来的对象,如果返回 NULL 的话,说明堆空间不够了

2.2.2 设置选项列表

```
void lv_ddlist_set_options(lv_obj_t * ddlist, const char * options);
```

参数:

ddlist: 下拉列表框对象

options: 选项列表,每一个选项值之间需要用'\n'换行符来作为区分,如 `const char * options = "First\nSecond\nThird";`就定义了 First, Secode, Third 这三个选项值

2.2.3 选中某个选项值

```
void lv_ddlist_set_selected(lv_obj_t * ddlist, uint16_t sel_opt);
```

参数:

ddlist: 下拉列表框对象

sel_opt: 选项索引值,从 0 开始

2.2.4 设置下拉列表框的高度

```
void lv_ddlist_set_fix_height(lv_obj_t * ddlist, lv_coord_t h);
```

参数:

ddlist: 下拉列表框对象

h: 高度值

如果不设置高度的话,也是没有关系的,下拉列表框控件会根据选项的个数来自动确定高度的

2.2.5 设置下拉列表框的宽度

```
void lv_ddlist_set_fix_width(lv_obj_t * ddlist, lv_coord_t w);
```

参数:

ddlist: 下拉列表框对象

w: 宽度值

如果不设置宽度的话,也是没有关系的,下拉列表框控件会根据选项值的最大文本长度来自动确定宽度的

2.2.6 是否使能绘制向下箭头

```
void lv_ddlist_set_draw_arrow(lv_obj_t * ddlist, bool en);
```

参数:

ddlist: 下拉列表框对象

en: 是否使能绘制收缩状态下右边的向下箭头

2.2.7 是否禁止自动返回到收缩状态

```
void lv_ddlist_set_stay_open(lv_obj_t * ddlist, bool en);
```

参数:

ddlist: 下拉列表框对象

en: true 代表禁止自动返回到收缩状态,false 代表不禁止

默认情况下是当用户从展开状态下选择好选项值时,下拉列表框会自动返回到收缩状态的,但是如果你调用 lv_ddlist_set_stay_open(ddlist,true)接口之后,它将会一直保持在展开状态,而不会自动返回到收缩状态了

2.2.8 设置滚动条的模式

```
static inline void lv_ddlist_set_sb_mode(lv_obj_t * ddlist, lv_sb_mode_t mode);
```

参数:

ddlist: 下拉列表框对象

mode: 滚动条的模式

此 API 接口的使用方法和 lv_page 页面章节中的 lv_page_set_sb_mode 接口的使用方法是相同的

2.2.9 设置动画时长

```
static inline void lv_ddlist_set_anim_time(lv_obj_t * ddlist, uint16_t anim_time);
```

参数:

ddlist: 下拉列表框对象

anim_time: 展开和收缩动画的时长,单位 ms

2.2.10 设置样式

```
void lv_ddlist_set_style(lv_obj_t * ddlist, lv_ddlist_style_t type, const lv_style_t * style);
```

参数:

ddlist: 下拉列表框对象

type: 设置哪一部分的样式,有如下 3 个可选值

LV_DDLIST_STYLE_BG: 修饰下拉列表框控件的背景

LV_DDLIST_STYLE_SEL: 修饰被选择的选项值

LV_DDLIST_STYLE_SB: 修饰滚动条

style: 样式

2.2.11 设置文本的水平对齐方式

```
void lv_ddlist_set_align(lv_obj_t * ddlist, lv_label_align_t align);
```

参数:

ddlist: 下拉列表框对象

align: 水平对齐方式

2.2.12 获取当前选择的文本内容

```
void lv_ddlist_get_selected_str(const lv_obj_t * ddlist, char * buf, uint16_t buf_size);
```

参数:

ddlist: 下拉列表框对象

buf: 用来存放选项值的文本内容

buf_size: buf 缓冲区的大小

2.2.13 获取当前选择的选项值的索引

```
uint16_t lv_ddlist_get_selected(const lv_obj_t * ddlist);
```

参数:

ddlist: 下拉列表框对象

返回值:

返回当前选择的选项值的索引值,从 0 开始的

2.2.14 将下拉列表框处于展开状态

```
void lv_ddlist_open(lv_obj_t * ddlist, lv_anim_enable_t anim);
```

参数:

ddlist: 下拉列表框对象

anim: 是否使能动画效果,有如下两个可选值

LV_ANIM_ON: 使能动画效果

LV_ANIM_OFF: 不使能动画效果

2.2.15 将下拉列表框处于收缩状态

```
void lv_ddlist_close(lv_obj_t * ddlist, lv_anim_enable_t anim);
```

参数:

ddlist: 下拉列表框对象

anim: 是否使能动画效果,有如下俩个可选值

LV_ANIM_ON: 使能动画效果

LV_ANIM_OFF: 不使能动画效果

2.2.16 备注

还有几个 get 获取类型的 API 接口我这里就不列举出来了,比较简单的

3. 例程设计

3.1 功能简介

创建 2 个自定义样式,分别用来修饰下拉列表框的背景和选项值被选中时的样式,接着创建一个下拉列表框对象,设置它的固定宽度,设置它的所有选项值,使能绘制向下箭头,最后给其设置事件回调函数,当按下 KEY0 按键时,会将下拉列表框强制设为展开状态,当按下 KEY1 按键时,会将下拉列表框强制设为收缩状态,当按下 WKUP 按键时,会禁止掉自动回到收缩状态的功能.

3.2 硬件设计

本例程所用到的硬件有:

- 1) 液晶屏
- 2) 串口
- 3) KEY0,KEY1,WKUP 按键

3.3 软件设计

在 GUI_APP 目录下创建 lv_ddlist_test.c 和 lv_ddlist_test.h 俩个文件,其中 lv_ddlist_test.c 文件的内容如下:

```
#include "lv_ddlist_test.h"
#include "lvgl.h"
#include "key.h"
#include <stdio.h>

//是否将下拉列表框的弹出方向改为向上,默认情况是向下的
#define DDLIST_DIRECTION_UP_EN    0

lv_style_t bg_style;
lv_style_t sel_style;
lv_obj_t * ddlist1;

//事件回调函数
static void event_handler(lv_obj_t * obj,lv_event_t event)
{
```



```
if(event == LV_EVENT_VALUE_CHANGED)
{
    char buf[32];
    uint16_t selected_index = lv_ddlist_get_selected(obj); //获取选项值的索引
    lv_ddlist_get_selected_str(obj, buf, sizeof(buf)); //获取选项值的文本内容
    printf("Option index: %d, Option text: %s\r\n", selected_index, buf);
}
}

//例程入口
void lv_ddlist_test_start()
{
    lv_obj_t *scr = lv_scr_act(); //获取当前活跃的屏幕对象

    //1. 创建样式
    //1.1 创建背景样式
    lv_style_copy(&bg_style, &lv_style_plain);
    bg_style.body.main_color = LV_COLOR_WHITE; //纯白色背景
    bg_style.body.grad_color = bg_style.body.main_color;
    bg_style.body.border.width = 1; //边框宽度
    bg_style.body.border.color = LV_COLOR_MAKE(0xAA, 0xAA, 0xAA);
    //LV_COLOR_MAKE(0x30, 0x30, 0x30);
    bg_style.body.padding.left = 10; //设置左侧的内边距
    bg_style.text.color = LV_COLOR_BLACK; //文本颜色
    bg_style.body.shadow.color = bg_style.body.border.color; //阴影颜色
    bg_style.body.shadow.width = 4; //阴影宽度

    //1.2 创建选择项被选中时的样式
    lv_style_copy(&sel_style, &lv_style_plain);
    sel_style.body.main_color = LV_COLOR_MAKE(0x5F, 0xB8, 0x78); //浅绿色背景
    sel_style.body.grad_color = sel_style.body.main_color;
    sel_style.text.color = LV_COLOR_WHITE; //文本为白色

    //2. 创建下拉列表框
    ddlist1 = lv_ddlist_create(scr, NULL);

    lv_ddlist_set_options(ddlist1, "Shanghai\nBeijing\nShenzhen\nGuangzhou\nHangzhou\nNanchang"); //设置列表选项
    lv_ddlist_set_selected(ddlist1, 4); //设置默认选中值为 Hangzhou
    lv_ddlist_set_fix_width(ddlist1, 140); //设置固定宽度
    lv_ddlist_set_draw_arrow(ddlist1, true); //使能绘制向下的箭头
    lv_ddlist_set_style(ddlist1, LV_DDLIST_STYLE_BG, &bg_style); //设置背景样式
```

```
lv_ddlist_set_style(ddlist1, LV_DDLIST_STYLE_SEL, &sel_style); // 设置背景样式
lv_obj_set_event_cb(ddlist1, event_handler); // 注册事件回调函数
lv_obj_align(ddlist1, NULL, LV_ALIGN_IN_TOP_MID, 0, 20); // 设置与屏幕的对齐方式
#if (DDLST_DIRECTION_UP_EN)

    // 重新设置与屏幕的对齐方式, 必须为底部对齐
    lv_obj_align(ddlist1, NULL, LV_ALIGN_IN_BOTTOM_MID, 0, -20);
    // 必须使能自动对齐, 只有满足这两点, 才能看到向上弹出的效果
    lv_obj_set_auto_realign(ddlist1, true);
#endif
}

// 按键处理
void key_handler()
{
    u8 key = KEY_Scan(0);

    if (key == KEY0_PRES)
    {
        lv_ddlist_open(ddlist1, LV_ANIM_ON); // 把下拉列表框强制设置为展开状态
    } else if (key == KEY1_PRES)
    {
        lv_ddlist_close(ddlist1, LV_ANIM_ON); // 把下拉列表框强制设置为收缩状态
    } else if (key == WKUP_PRES)
    {
        lv_ddlist_set_stay_open(ddlist1, true); // 禁止自动返回到收缩状态
    }
}
```

3.4 下载验证

把代码下载进去之后,可以看到如下所示的初始界面效果:



图 3.4.1 初始界面效果

接着我们用手点击一下此下拉列表框,会看到如下展开状态下的效果:



图 3.4.2 展开状态下的界面效果

最后我们还可以按 KEY0, KEY1, WKUP 等按键来测试其他的功能.

4. 资料下载

正点原子公司名称：广州市星翼电子科技有限公司

LittleVGL 资料连接：www.openedv.com/thread-309664-1-1.html

原子哥在线教学平台：www.yuanzige.com

正点原子淘宝店铺：<https://openedv.taobao.com>

正点原子官方网站：www.alientek.com

正点原子 B 站视频：<https://space.bilibili.com/394620890>

电话：020-38271790 传真：020-36773971

请下载原子哥 APP，数千讲视频免费学习，更快更流畅。

请关注正点原子公众号，资料发布更新我们会通知。



扫码下载“原子哥”APP



扫码关注正点原子公众号