

LittlevGL documentation (Turkish)

Table of contents

[Hoşgeldiniz](#)

[Porting](#)

[PC Simulator](#)

[Nesneler](#)

[Biçimler](#)

[Giriş Aygıtları](#)

[Renkler](#)

[Fontlar](#)

[Çizimler](#)

[Animasyon](#)

[Kodlama Kılavuzu](#)

[Nesne Türleri](#)

[Anahtar \(lv_sw\)](#)

[Açılır liste \(lv_ddlist\)](#)

[Buton \(lv_btn\)](#)

[Buton matrisi \(lv_btnm\)](#)

[Çizge \(lv_chart\)](#)

[Çizgi \(lv_line\)](#)

[Çizgi ölçer \(lv_lmeter\)](#)

[Etiket \(lv_label\)](#)

[Kaydırıcı \(lv_slider\)](#)

[Klavye \(lv_kb\)](#)

[Konteynır \(lv_cont\)](#)

[LED \(lv_led\)](#)

[Liste \(lv_list\)](#)

[Mesaj kutusu \(lv_mbox\)](#)

[Onay kutusu \(lv_cb\)](#)

[Önyükleyici \(lv_preload\)](#)

[Pencere \(lv_window\)](#)

[Resim \(lv_img\)](#)

[Resim buton \(lv_imgbtn\)](#)

[Sarma çubuğu \(lv_roller\)](#)

[Sayfa \(lv_page\)](#)

[Sekme görünümü \(lv_tabview\)](#)

[Sütun \(lv_bar\)](#)

[Takvim \(lv_calendar\)](#)

[Tartı \(lv_gauge\)](#)

[Temel nesne \(lv_obj\)](#)

[Yay \(lv_arc\)](#)

[Yazı alanı \(lv_ta\)](#)

Hoşgeldiniz

Written for v5.3, revision 2



Welcome LittlevGL Embedded GUI Library



LittlevGL is a free and open-source graphics library providing everything you need to create embedded GUI with easy-to-use graphical elements, beautiful visual effects and low memory footprint.

Key features

- **Powerful building blocks** buttons, charts, lists, sliders, images, etc.
- **Advanced graphics** with animations, anti-aliasing, opacity, smooth scrolling
- **Various input devices** touch pad, mouse, keyboard, encoder, buttons, etc.
- **Multi-language support** with UTF-8 encoding
- **Fully customizable** graphical elements
- **Hardware independent** to use with any microcontroller or display
- **Scalable** to operate with little memory (50 kB Flash, 10 kB RAM)
- **OS, External memory and GPU** supported but not required
- **Single frame buffer** operation even with advances graphical effects
- **Written in C** for maximal compatibility
- **Simulator** to develop on PC without embedded hardware
- **Tutorials, examples, themes** for rapid development

How to get started?

Read the documentation

Reading the documentation first is always a good idea. Don't be afraid it takes only a little time to learn the basics.

You can start here with [Porting](#) or with an [Introduction](#) to the library.

Tutorial

There is a tutorial to show you the most important parts step-by-step.

It's located in the [lv_examples repository](#)

Try LittlevGL in a PC simulator

If you don't have an embedded hardware with a display yet you can try the library on PC. The PC simulator uses a window on your monitor to simulate a display and uses your mouse to click on that display.

The simulator works with Windows, Linux, and OSX as well.

Here you can learn how to set-up the simulator: [PC simulator](#)

Contribute

Use GitHub's issue tracker to:

- Report bugs
- Suggest new features
- Add new features
- Help others

Before contributing, read the related document: [CONTRIBUTING](#)

Pencere (lv_window)

Written for v5.1 (rev.4)

Genel Bakış

Pencereler, **en karmaşık** konteyner benzeri nesnelerden biridir. İki ana bölümden oluşur: Üstte bir **başlık Konteyner** ve başlığın altındaki **içerik** için bir [Sayfa](#).

Üstbilgi **başlığı** değiştirilebilir: `lv_win_set_title(win, "Yeni Başlık")`. Başlık, üstbilginin biçimini her zaman miraslanır.

Üst bilginin sağına **kontrol butonları** `lv_win_add_btn(win, "U:/close", my_close_action)` fonksiyonu ile eklenebilir. İkinci parametre bir resim dosyası yoludur, üçüncü parametre buton bırakıldığı zaman çağrılan bir fonksiyondur. **Semboller** resim gibi kullanabilir: `lv_win_add_btn(win, SYMBOL_CLOSE, my_close_action)`.

Kontrol butonlarının boyutu `lv_win_set_btn_size(win, yeni_boyut)` fonksiyonuyla değiştirilebilir.

Kaydırma çubuğunun davranışı `lv_win_set_sb_mode(win, LV_SB_MODE_...)` fonksiyonuyla ayarlanabilir.

İçerik düzenini ayarlamak için `lv_win_set_layout(win, LV_LAYOUT_...)` fonksiyonu kullanılır.

Biçim kullanımı

Pencerenin ögesine yeni bir stil ayarlamak için `lv_win_set_style(win, LV_WIN_STYLE_..., &style)` fonksiyonu kullanılır.

- **LV_WIN_STYE_BG** tüm *style.body* özelliklerini kullanan ana arka plan(üstbilgi ve içerik yer alır) (varsayılan: lv_style_plain)
- **LV_WIN_STYLE_CONTENT_BG** tüm *style.body* özelliklerini kullanan içerik sayfasının arka planı (varsayılan: lv_style_transp)
- **LV_WIN_STYLE_CONTENT_SCRL** tüm *style.body* özelliklerini kullanan içerik sayfasının kaydırma çubuğu (varsayılan: lv_style_transp)
- **LV_WIN_STYLE_SB** tüm *style.body* özelliklerini kullanan kaydırma çubuğunun biçimi. Kaydırma çubuğunun dolgusu sırasıyla yatay/dikey* dolgularıyla ayarlanır ve kaydırma çubuğunun genişliği de iç dolguyla ayarlanır. (varsayılan: lv_style_pretty_color)
- **LV_WIN_STYLE_HEADER** tüm *style.body* özelliklerini kullanan üstbilgi biçimi (varsayılan: lv_style_plain_color)
- **LV_WIN_STYLE_BTN_REL** tüm *style.body* özelliklerini kullanan butonu bırakman(üstbilgi) biçimi (varsayılan: lv_style_btn_rel)
- **LV_WIN_STYLE_BTN_PR** tüm *style.body* özelliklerini kullanan butona basma(üstbilgi) biçimi (default: lv_style_btn_pr)

Notlar

Örnek

□

```

/*Yeni bir kaydırma çubuğu biçimi oluştur*/
static lv_style_t style_sb;
lv_style_copy(&style_sb, &lv_style_plain);
style_sb.body.main_color = LV_COLOR_BLACK;
style_sb.body.grad_color = LV_COLOR_BLACK;
style_sb.body.border.color = LV_COLOR_WHITE;
style_sb.body.border.width = 1;
style_sb.body.border.opa = LV_OPA_70;
style_sb.body.radius = LV_RADIUS_CIRCLE;
style_sb.body.opa = LV_OPA_60;

/*Yeni bir pencere oluştur*/
lv_obj_t * win = lv_win_create(lv_scr_act(), NULL);
lv_win_set_title(win, "Example window");
lv_win_set_style(win, LV_WIN_STYLE_SB, &style_sb);

/*Başlığı ayarla*/
/*Kaydırma çubuğunun biçimini ayarla*/

/*Üstbilgiye kontrol butonu ekle*/
lv_win_add_btn(win, SYMBOL_SETTINGS, my_setup_action);
lv_win_add_btn(win, SYMBOL_CLOSE, lv_win_close_action);

/*Ayar butonu ekle*/
/*Kapatma butonu ekle ve kapatma aksiyonunun kullanılması*/

/*İçerik örneği ekle*/
lv_obj_t * txt = lv_label_create(win, NULL);
lv_label_set_text(txt, "This is the content of the window\n\n"
    "You can add control buttons to\nthe window header\n\n"
    "You can scroll it\n\n"
    "See the scroll bar on the right!");

```

Sütun (lv_bar)

Written for v5.1

Genel Bakış

Bar nesnesi **arkaplan** ve **işaretleyici(gösterge)** olmak üzere iki parçadan oluşur. Nesnenin temeli **arkaplandır**, **işaretleyici(gösterge)** arkaplan ile şekil olarak benzerdir fakat genişlik ve yüksekliği ayarlanabilir.

Bar'ın yönü genişlik/yükseklik oranına bağlı olarak **dikey veya yatay** olabilir. Mantıksal olarak yatay barlar genişlik, dikey barlar yükseklik değerleri değiştirilerek oluşturulur.

`lv_bar_set_value(bar, new_value)` fonksiyonu ile **yeni değer** atanır. Bu değer `lv_bar_set_range(bar, min, max)` fonksiyonu ile belirlenmiş **aralık** (minimum ve maximum değerler) arasında bir değerdir. Varsayılan aralık değerleri 1..100 arasındadır.

Güncel değerden istenilen değere geçiş **animasyon** lu biçimde olur. Bu durumda `lv_bar_set_value_anim(bar, yeni_deger, animasyon_zamani)` fonksiyonu kullanılır.

Biçim Kullanımı

- Nesnenin temeli arkaplan** bu nedenle biçim elementleri kullanılır. Varsayılan biçim `LV_STYLE_PRETTY` dir.
- işaretleyici(gösterge)** arkaplanın benzeridir. İşaretleyici(gösterge) biçimleri `lv_bar_set_style_indic(bar, &style_indic)` fonksiyonu ile ayarlanır ve arkaplan ile arasında mesafe bırakmak için *hpad* and *vpad* biçim elementleri kullanılır. Varsayılan biçim olarak `LV_STYLE_PRETTY_COLOR` kullanılır.

Notlar

- İşaretleyici gerçek bir nesne değildir; sadece bar tarafından çizilmiştir.

Örnek

□

```

/*Varsayılan Bar oluştur*/
lv_obj_t * bar1 = lv_bar_create(lv_scr_act(), NULL);
lv_obj_set_size(bar1, 200, 30);
lv_obj_align(bar1, NULL, LV_ALIGN_IN_TOP_RIGHT, -20, 30);
lv_bar_set_value(bar1, 70);

/*Bar'ın sağında etiket oluştur*/
lv_obj_t * bar1_label = lv_label_create(lv_scr_act(), NULL);
lv_label_set_text(bar1_label, "Default");
lv_obj_align(bar1_label, bar1, LV_ALIGN_OUT_LEFT_MID, -10, 0);

/*Bar ve işaretleyici(gösterge) için biçim oluştur*/
static lv_style_t style_bar;
static lv_style_t style_indic;

lv_style_copy(&style_bar, &lv_style_pretty);
style_bar.body.main_color = LV_COLOR_BLACK;
style_bar.body.grad_color = LV_COLOR_GRAY;
style_bar.body.radius = LV_RADIUS_CIRCLE;
style_bar.body.border.color = LV_COLOR_WHITE;

lv_style_copy(&style_indic, &lv_style_pretty);
style_indic.body.grad_color = LV_COLOR_GREEN;
style_indic.body.main_color = LV_COLOR_LIME;
style_indic.body.radius = LV_RADIUS_CIRCLE;
style_indic.body.shadow.width = 10;
style_indic.body.shadow.color = LV_COLOR_LIME;
style_indic.body.padding.hor = 3; /*İşaretleyici(gösterge)'yi biraz küçükt*/
style_indic.body.padding.ver = 3;

/*İkinci bar oluştur*/
lv_obj_t * bar2 = lv_bar_create(lv_scr_act(), bar1);
lv_bar_set_style(bar2, LV_BAR_STYLE_BG, &style_bar);
lv_bar_set_style(bar2, LV_BAR_STYLE_INDIC, &style_indic);
lv_obj_align(bar2, bar1, LV_ALIGN_OUT_BOTTOM_MID, 0, 30); /*'Bar1'in altında hizala*/

/*Bar'ın sağında ikinci bir etiket oluştur*/
lv_obj_t * bar2_label = lv_label_create(lv_scr_act(), bar1_label);
lv_label_set_text(bar2_label, "Modified");
lv_obj_align(bar2_label, bar2, LV_ALIGN_OUT_LEFT_MID, -10, 0);

```


Yay (lv_arc)

Written for v5.2

Genel Bakış

Belirlenen **kalınlık** ile **başlangıç ve bitiş** açıları verilmiş bir **Yay nesnesi** çizilmesi.

Açıların ayarlanması için `lv_arc_set_angles(arc(yay), baslangic_acisi, bitis_acisi)` fonksiyonu kullanılır. Sıfır derece nesnenin altından başlar ve saat tersi yönünde devam eder, açı değerleri [0;360] arasındadır.

Yay nesnesinin biçim(style) ayarlaması için `lv_arc_set_style(arc, LV_ARC_STYLE_MAIN, &style)` fonksiyonu kullanılır.

Biçim kullanımı

- **line.rounded** bitiş noktasını yuvarlaştırmak için kullanılır (Şeffaflık(opacity) 1 yapılırsa bu özellik çalışmaz)
- **line.width** Yay kalınlığı.
- **line.color** Yay rengi.

Notlar

- Yay için **genişlik ve boy** aynı olmalı.
- Yay **anti-aliasing** i şuan **desteklememektedir**.

Örnek

□

```
/*Yay için biçim oluştur*/
lv_style_t style;
lv_style_copy(&style, &lv_style_plain);
style.line.color = LV_COLOR_BLUE;           /*Yay rengi*/
style.line.width = 8;                       /*Yay kalınlığı*/

/*Yay oluştur*/
lv_obj_t * arc = lv_arc_create(lv_scr_act(), NULL);
lv_arc_set_style(arc, LV_ARC_STYLE_MAIN, &style);           /*Yeni biçim kullanımı*/
lv_arc_set_angles(arc, 90, 60);
lv_obj_set_size(arc, 150, 150);
lv_obj_align(arc, NULL, LV_ALIGN_CENTER, 0, 0);

/*Bir önceki yay'ı kopyalayıp farklı açılarda ve boyutta yay oluştur*/
arc = lv_arc_create(lv_scr_act(), arc);
lv_arc_set_angles(arc, 90, 20);
lv_obj_set_size(arc, 125, 125);
lv_obj_align(arc, NULL, LV_ALIGN_CENTER, 0, 0);

/*Bir önceki yay'ı kopyalayıp farklı açılarda ve boyutta yay oluştur*/
arc = lv_arc_create(lv_scr_act(), arc);
lv_arc_set_angles(arc, 90, 310);
lv_obj_set_size(arc, 100, 100);
lv_obj_align(arc, NULL, LV_ALIGN_CENTER, 0, 0);
```

Yazı alanı (lv_ta)

Written for v5.1 (rev.3)

Genel Bakış

Metin alanı, üzerinde **etiket** ve **imleç** bulunan bir **sayfadır**. İmleğin bulunduğu konuma **metin veya karakter girerek** eklenir:

- `lv_ta_add_char(ta, 'c')`
- `lv_ta_add_text(ta, "insert this text")`

`lv_ta_set_text(ta, "New text")` fonksiyonu **tüm metni değiştirir**

Geçerli imleç pozisyonunun solundan **bir karakter silmek** için `lv_ta_del()` kullanılır.

İmleç pozisyonu doğrudan `lv_ta_set_cursor_pos(ta, 10)` fonksiyonuyla ya da adım atılarak değiştirilebilir:

- `lv_ta_cursor_right(ta)`
- `lv_ta_cursor_left(ta)`
- `lv_ta_cursor_up(ta)`
- `lv_ta_cursor_down(ta)`

Birkaç farklı imleç türü vardır. Bunlardan biri şu şekilde ayarlanabilir: `lv_ta_set_cursor_type(ta, LV_CURSOR_...)`

- LV_CURSOR_NONE
- LV_CURSOR_LINE
- LV_CURSOR_BLOCK
- LV_CURSOR_OUTLINE
- LV_CURSOR_UNDERLINE

İmleci gizlemek için 'OR' LV_CURSOR_HIDDEN_ kullanılabilir.

Metin alanı `lv_ta_set_one_line(ta, true)` fonksiyonu ile çizgili olacak şekilde yapılandırılır.

Metin alanı **şifre modunu** destekler. `lv_ta_set_pwd_mode(ta, true)` fonksiyonu ile etkinleştirilir.

Biçim kullanımı

Metin alanının bir ögesine yeni bir biçim ayarlamak için `lv_ta_set_style(page, LV_TA_STYLE_..., &style)` fonksiyonu kullanılır:

- **LV_TA_STYLE_BG** tüm *style.body* özelliklerini kullanan arka plan biçimi. Ayrıca etiket de *style.label* 'den bu biçimi kullanır. (varsayılan: lv_style_pretty)
- **LV_TA_STYLE_SB** tüm *style.body* özelliklerini kullanan kaydırma çubuğunun biçimi (varsayılan: lv_style_transp)
- **LV_TA_STYLE_CURSOR** imleç biçimi. NULL ise kütüphane, etiket rengine ve fontuna göre otomatik olarak biçim ayarlar.
 - LV_CURSOR_LINE: *style.line.width* geniş bir çizgi ama *style.body* olarak bir dikdörtgen olarak çizilir. Yatay ve dikey dolgu imleç üzerinde ofset yapar.
 - LV_CURSOR_BLOCK: *style.body* olarak bir dikdörtgen, yatay ve dikey olarak dolguyla dikdörtgeni büyütür.
 - LV_CURSOR_OUTLINE: *style.body* olarak boş bir dikdörtgen(sadece kenarlık), yatay ve dikey olarak dolguyla dikdörtgeni büyütür.
 - LV_CURSOR_UNDERLINE: *style.line.width* geniş çizgi ancak *style.body* şeklinde bir dikdörtgen çizildiğinde imleç üzerinde ofset yapar.

Notlar

- Şifre modunda `lv_ta_get_text(ta)` yıldız karakterlerini değil gerçek metni verir.

Örnek

□

```
/*Kaydırma çubuğunun biçimini ayarla*/
static lv_style_t style_sb;
lv_style_copy(&style_sb, &lv_style_plain);
style_sb.body.main_color = LV_COLOR_BLACK;
style_sb.body.grad_color = LV_COLOR_BLACK;
style_sb.body.border.color = LV_COLOR_WHITE;
style_sb.body.border.width = 1;
style_sb.body.border.opa = LV_OPA_70;
style_sb.body.radius = LV_RADIUS_CIRCLE;
style_sb.body.opa = LV_OPA_60;

/*Normal metin alanı oluştur*/
lv_obj_t * ta1 = lv_ta_create(lv_scr_act(), NULL);
lv_obj_set_size(ta1, 200, 100);
lv_obj_align(ta1, NULL, LV_ALIGN_CENTER, 0, - LV_DPI / 2);
lv_ta_set_style(ta1, LV_TA_STYLE_SB, &style_sb);
lv_ta_set_cursor_type(ta1, LV_CURSOR_BLOCK);
lv_ta_set_text(ta1, "A text in a Text Area\n"
                  "You can scroll it if the text is long enough.");
lv_ta_set_cursor_pos(ta1, 2);
lv_ta_add_text(ta1, "long ");

/*Kaydırma çubuğu biçimini uygula*/
/*İlk metni ayarla*/
/*İmleç konumunu ayarla*/
/*Geçerli imleç konumuna bir kelime ekle*/

static lv_style_t style_bg;
lv_style_copy(&style_bg, &lv_style_pretty);
style_bg.body.shadow.width = 8;
style_bg.text.color = LV_COLOR_MAKE(0x30, 0x60, 0xd0);

/*Mavi etiket*/

/*Şifre modu ile tek çizgili bir test oluştur*/
lv_obj_t * ta2 = lv_ta_create(lv_scr_act(), ta1);
lv_obj_align(ta2, ta1, LV_ALIGN_OUT_BOTTOM_MID, 0, 50);
lv_ta_set_style(ta2, LV_TA_STYLE_BG, &style_bg);
lv_ta_set_one_line(ta2, true);
lv_ta_set_cursor_type(ta2, LV_CURSOR_LINE);
lv_ta_set_pwd_mode(ta2, true);
lv_ta_set_text(ta2, "Password");

/*Arkaplan biçimini uygula*/
```