

## Класс TEv3Leds, модуль rubiroLeds

```
TEv3Leds=class(TEv3)
  constructor create();
  destructor destroy; override;
  //запускается при старте, создает светло-оранжевый цвет
  function start:TEv3Leds;
  // запускается по завершении, ввостанавливает штанный зеленый
  function stop:TEv3Leds;
  // сохраняет текущие цвета в памяти (на вершине стека)
  function push:TEv3Leds;
  // ввостанавливает цвета из памяти (с вершины стека)
  function pop:TEv3Leds;
  // гасит все цвета
  function hide:TEv3Leds;
  {
    при последовательном использовании гасит-ввостанавливает цвета
    применяется для имитация мигания
    не рекомендуется использовать совместно с
    push, pop, hide - возможны аберрации
  }
  function blink:TEv3Leds;
  {
    Если в цвета были погашены миганием, ввостанавливает их,
    иначе ничего не делает
  }
  function unblink:TEv3Leds;
  {
    Устанавливает интенсивность всех цветоиндикаторов
    диапазон значений - от 0 до 255
    (от минимальной до максимальной интенсивности)
    значение -1 означает оставить соответствующий цвет без изменений
  }
  function All(_leftred,_leftgreen,_rightred,_rightgreen:variant):TEv3Leds;
  // устанавливает интенсивность левых цветоиндикаторов
  function Left(red,green:variant):TEv3Leds;
  // устанавливает интенсивность правых цветоиндикаторов
  function Right(red,green:variant):TEv3Leds;
  // устанавливает интенсивность левого красного цветоиндикатора
  function LeftRed(red:variant):TEv3Leds;
  // устанавливает интенсивность правого красного цветоиндикатора
  function RightRed(red:variant):TEv3Leds;
  // устанавливает интенсивность левого зеленого цветоиндикатора
  function LeftGreen(green:variant):TEv3Leds;
  // устанавливает интенсивность правого зеленого цветоиндикатора
  function RightGreen(green:variant):TEv3Leds;
end;
```

EV3 имеет 4 цветоиндикатора: левый красный, левый зеленый, правый красный и правый зеленый. При смешении цветов индикаторов одной стороны можно получать различные оттенки зеленого, красного и оранжевого цветов.

Интенсивность цвета определяется целым значением от 0 до 255.

При запуске программы автоматически вызывается метод start, устанавливающий светло-оранжевый цвет на цветоиндикаторах. Используется как визуальное подтверждение старта программы и успешной инициализации библиотеки. При завершении вызывается метод stop, восстанавливающий стандартный зеленый цвет на цветоиндикаторах.

Методы push и pop позволяют сохранять и восстанавливать все значения цветоиндикаторов одновременно. Для этого используется стек, размер которого ограничен только доступной оперативной памятью. Метод blink использует тот-же стек для имитации мигания и применяется, например, в методах waitAny и waitAll базового класса TEv3 (примеры применения - см. документацию по классу TEv3Buttons)

При установке цвета каждого индикатора используется повышенный таймаут - 30мс. Практика показывает, что при меньшем таймауте драйвер цветоиндикатора может "захлебнуться" быстро поступающими данными, что приведет к зависанию контроллера.

Примеры:

```
// троекратное "переливание" цветов (led1.pp)
{$mode objfpc}
uses uev3, rubiroLeds;
var i, k: integer;
begin
  ev3Init();
  for k:=1 to 3 do begin
    for i:=0 to 255 do begin
      ev3Leds.all(255-i, i, i, 255-i);
    end;
    for i:=0 to 255 do begin
      ev3Leds.all(i, 255-i, 255-i, i);
    end;
  end;
end.

// демонстрация мигания цветоиндикаторов (led2.pp)
// при отпущенной центральной кнопке - мигание оранжевым
// при нажатой центральной кнопке - мигание красным
// выход из программы - нажатие на кнопку Back
{$mode objfpc}
uses uev3, rubiroLeds, rubiroButtons, sysutils;
var
  centered: boolean = false;
begin
  ev3Init();
  while true do begin
    if ev3buttons.center and not centered then begin
      centered := true;
      ev3Leds.unblink.all(255, 0, 255, 0);
    end;
    if not ev3buttons.center and centered then begin
      centered := false;
      ev3Leds.unblink.start;
    end;
    if ev3Buttons.back then break;
    ev3Leds.blink();
    sleep(300);
  end;
end.
```