

ssd1306IIC

AVR GCC OLED 128x64 with ssd1306 controller display
library

ИНСТРУКЦИЯ

Введение:

Эта библиотека написана для OLED дисплеев разрешением 128x64, контроллером ssd1306 и интерфейсом I2C. Библиотека использует аппаратный интерфейс TWI микроконтроллеров AVR для реализации протокола I2C.

Основное отличие от большинства библиотек заключается в отслеживании размеров измененной области буфера дисплея в оперативной памяти, и ускоренная за счет этого перерисовка изображения на дисплее.

Последнюю версию Вы можете найти на сайте <http://kot3245.ru>

Определенные перечисления:

enum OLEDpixelMode

Используется с функцией **setPixel()**;

Возможные значения: ON/OFF/XOR соответственно для включения/выключения/инверсии пикселя

enum OLEDalign

Используется с функцией **str()**;

Возможные значения: LEFT/RIGHT/CENTER

Используется для выравнивания текста.

enum OLEDinverted

Используется с функциями **invert()**; **invertText()**;

Возможные значения: NORMAL/INVERT

Функции:

void initSSD1306(void);

Описание: Инициализирует дисплей.

Аргументы: нет.

Использование: **initSSD1306();**

void setBrightness(uint8_t value);

Описание: Устанавливает яркость дисплея.

Аргументы: **uint8_t value (0...255)** - значение яркости дисплея.

Использование: **void setBrightness(201);** //устанавливает значение яркости в 201.

void update(void);

Описание: Перерисовывает измененную другими функциями область дисплея. Большинство функций библиотеки требует вызов этой функции для отображения результата их работы. Исключение составляют функции: `setBrightness()`; `invert()`; Остальные функции библиотеки работают с буфером в RAM.

Аргументы: нет.

Использование: `update();` //перерисовывает измененную область дисплея.

void clrScr(OLEDpixelMode mode);

Описание: Очищает или заполняет буфер экрана единицами.

Аргументы: `OLEDpixelMode mode (NORMAL/INVERT)` - очистить или заполнить буфер дисплея значением `0xFF`.

Использование: `clrScr (NORMAL);` // очищает буфер дисплея

void chgPixel(uint8_t x, uint8_t y, OLEDpixelMode mode);

Описание: Изменяет один пиксель в RAM.

Аргументы: `uint8_t x (0...127)` - координата x пикселя;

`uint8_t y (0...63)` - координата y пикселя;

`OLEDpixelMode mode (ON/OFF/XOR)` - режим изменения пикселя позволяет его Включить/Выключить/Инвертировать;

Использование: `chgPixel (10,11,XOR);` //инвертирует пиксель по координатам (10,11)

void invert(OLEDInverted mode);
Описание: Устанавливает Нормальный/Инверсный режим отображения дисплея. Аргументы: OLEDInverted mode (NORMAL/INVERT). Использование: invert(INVERT); // устанавливает инверсный режим отображения дисплея.
void rInvertStr(uint8_t ry);
Описание: Устанавливает Нормальный/Инверсный режим отображения строки. Аргументы: uint8_t ry (0...) - задает относительный номер инвертируемой строки. Нумерация строк начинается с 0. Количество строк зависит от высоты используемого шрифта. Использование: invertStr(2); // устанавливает инверсный режим отображения для третьей сверху строки.
void invertText(OLEDInverted mode);
Описание: Устанавливает Нормальный/Инверсный режим отображения шрифта выводимого функциями chr(), rchr(), str(), rstr(). Аргументы: OLEDInverted mode (NORMAL/INVERT) - указывает режим отображения используемого шрифта. Использование: invertText(INVERT); // включает инверсный режим отображения шрифта.
void setFont(const uint8_t* font);
Описание: Устанавливает шрифт используемый функциями chr(), rchr(), str(), rstr(). Аргументы: const uint8_t* font - указатель на используемый шрифт. Использование: setFont(MediumNumbers);

```
void chr(unsigned char c, uint8_t x, uint8_t y);
```

Описание: выводит символ по абсолютным координатам дисплея

Аргументы: **unsigned char c** - выводимый символ.

uint8_t x (0...127) - координата x

uint8_t y (0...63) - координата y

Использование: **chr ('Z', 21,12);** // выводит символ Z по координатам (21,12)

```
void rchr(unsigned char c, uint8_t rx, uint8_t ry);
```

Описание: выводит символ по относительным координатам дисплея

Аргументы: **unsigned char c** - выводимый символ.

uint8_t x (0...) - относительная координата x

uint8_t y (0...) - относительная координата y

Использование: **chr ('Z', 3, 2);** // выводит символ Z в позицию 4 строки 3

```
void str(char *st, OLEDalign align, uint8_t y);
```

Описание: выводит строку по абсолютным координатам дисплея

Аргументы: **char *st** - указатель на выводимую строку.

OLEDalign align (LEFT/RIGHT/CENTER) -

 выравнивание строки

uint8_t y (0...63) - координата y

Использование: **str ("Hello world!", 20, CENTER);** // выводит строку "Hello world!" со смещением по y = 20, и выравниванием строки по центру.

```
void rstr (char *st, uint8_t rx, uint8_t ry);
```

Описание: выводит строку по относительным координатам дисплея

Аргументы: **char *st** - указатель на выводимую строку.

uint8_t x (0...) - относительная координата x

uint8_t y (0...) - относительная координата y

Использование: **rstr ("Hello world!", 0, 2);** // выводит строку "Hello world!" в позицию первого символа в третью строку.

void image(const unsigned char *imageData);

Описание: Копирует изображение из памяти RAM в буфер дисплея. Размер изображения 128x64 пикселя.

Аргументы: **const unsigned char *imageData** - указатель на изображение в RAM.

Использование: **image (kot3245);** // kot3245 - указатель на изображение сохраненное в PROGMEM

ssd1306IIC

Версии документа:

13.07.2017 - 1.0 Первоначальная версия

автор kot3245(at)ya.ru

сайт <http://kot3245.ru>