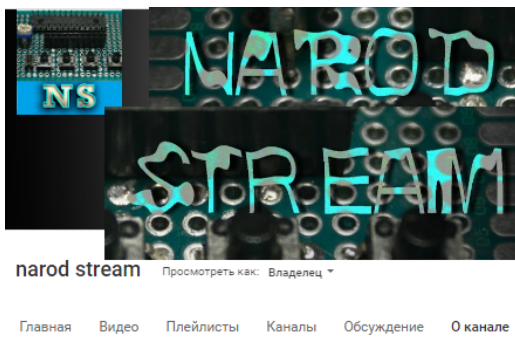


Сайт narodstream.ru

создан в поддержку
канала YouTube

NAROD STREAM







Рубрики


- [Uncategorized](#)
- [Программирование AVR](#)
- [Программирование STM32](#)

Свежие записи

- [STM32. Урок 94. DS18B20. Несколько датчиков на одной шине. Часть 1](#)
- [Ограничение доступа к сайту за чрезмерную активность](#)
- [STM32. Урок 93. LAN. W5500. HTTP Server. Сокеты. Часть 2](#)
- [STM32. Урок 93. LAN. W5500. HTTP Server. Сокеты. Часть 1](#)
- [STM32. Урок 92. Датчик температуры DS18B20. Часть 3](#)

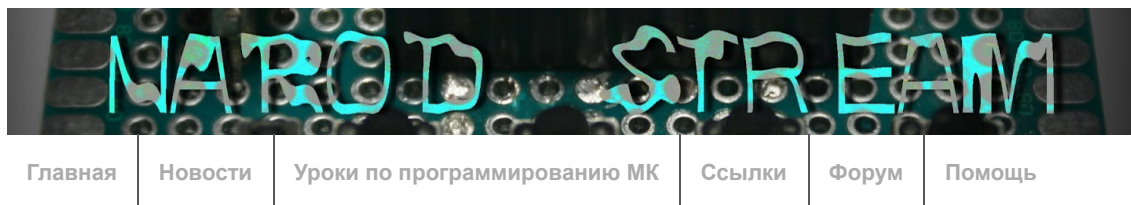
Последние ответы на форуме

-  [Narod Stream](#) в [Программирование МК STM32](#)
2 дн., 3 час. назад
-  [Mihail](#) в [Программирование МК STM32](#)
2 дн., 4 час. назад
-  [Dmitriy](#) в [Программирование МК AVR](#)
2 нед., 5 дн. назад
-  [nsk](#) в [Программирование МК STM32](#)
2 нед., 6 дн. назад

-  [Narod Stream](#) в [Программирование МК STM32](#)
3 нед. назад

Свежие комментарии

- [Narod Stream](#) к записи [AVR Урок 4](#).
[Смотрим результаты работы](#)
- [Narod Stream](#) к записи [STM Урок 44](#).
[SDIO. FATFS](#)
- [Виктор](#) к записи [AVR Урок 4](#). [Смотрим результаты работы](#)
- [Narod Stream](#) к записи [STM Урок 56](#).
[System Workbench. Подключаем библиотеку BSP. Часть 1](#)
- [3k](#) к записи [STM Урок 56](#). [System Workbench. Подключаем библиотеку BSP. Часть 1](#)



Просмотров: 37

[Главная](#) > [AVR Урок 34. Дисплей TFT 240×320 8bit. Часть 7](#)

Урок 34 Часть 7

Дисплей TFT 240×320 8bit

В [прошлой части](#) нашего урока мы написали ещё несколько функций для работы с дисплеем и вывели на его экран другие примитивы (линии, прямоугольники, окружности).

В данной заключительной части урока мы поработаем уже с текстом. Это будет нелёгкая работа, но зато успокаивает то что эта часть будет последняя в данном уроке.

В одной из предыдущих частей мы создали массив для шрифта размером в 16 пикселей, правда там не все символы, а только некоторые, так как размещение во флеш-памяти всех символов невозможно из-за её нехватки. Добавим также массив шрифта в 8 пикселей

Мета

- [Регистрация](#)
- [Войти](#)
- [RSS записей](#)
- [RSS комментариев](#)
- [WordPress.org](#)

Уроки по программированию МК

[Программирование МК AVR](#)

[Программирование МК STM32](#)

[Программирование МК PIC](#)

[Тесты устройств и аксессуаров](#)



Теперь добавим функцию по выводу символа определённой высоты в определённое место экрана

```
//-----
void TFT9341_Draw_Char(int x, int y, unsigned int
color, unsigned int phone, unsigned char charcode,
unsigned char size)
{
}
//-----
```

Во входных параметрах будут координаты вывода символа, цвет символа, цвет фона, код символа и размер символа.

Добавим возможность отслеживания размера с помощью операции ветвления

```
void TFT9341_Draw_Char(int x, int y, unsigned int
color, unsigned int phone, unsigned char charcode,
unsigned char size)
{
    switch(size)
    {
        int i,h;
        case 1:
            break;
        case 2:
            break;
        case 3:
            break;
        case 4:
            break;
    }
}
```

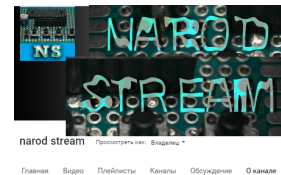
вариант 1 будет у нас для шрифта размером 8 пикселей, а вариант 2 для 16 пиксельного, остальные два варианта пока будут в запасе.

Начнем заполнять первый кейс, написав туда двухмерный цикл

```
case 1:
    for(h=0;h<8;h++)
    {
        for(i=0;i<8;i++)
        {
        }
    }
    break;
```

В принципе, несложно догадаться, что мы из элемента массива собираем пиксели по горизонтали (i) и по вертикали (h).

Заходите на канал Narod Stream



narod stream "Понимать и делать"

Главная Видео Плейлисты Каналы Обсуждение О канале

Архивы

- [Октябрь 2017](#)
- [Сентябрь 2017](#)

Теперь внутри внутреннего цикла будет условие, которое в зависимости от значения проверяемого бита в элементе массива будет отрисовывать точку определённого цвета. При единице будет цвет символа, а при нуле — цвет фона

```
for(i=0;i<8;i++)
{
    if ((pgm_read_byte(&chars8[charcode-0x20][h]))>>(7-i))&0x01)
    {
        TFT9341_DrawPixel(x+i,y+h,color);
    }
    else
    {
        TFT9341_DrawPixel(x+i,y+h,phone);
    }
}
```

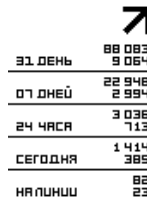
В случае второго варианта (размер шрифта 16 пикселей) будет также двухмерный цикл, только h уже будет доходить до 16, а не до восьми

```
case 2:
    for(h=0;h<16;h++)
    {
        for(i=0;i<8;i++)
        {
            // ...
        }
    }
    break;
```

Далее в теле внутреннего цикла у нас будет уже два условия, одно — для первого байта горизонтали, а второе — для второго

```
for(i=0;i<8;i++)
{
    if ((pgm_read_byte(&chars16[charcode-0x20][h*2]))>>(7-i))&0x01)
    {
        TFT9341_DrawPixel(x+i,y+h,color);
    }
    else
    {
        TFT9341_DrawPixel(x+i,y+h,phone);
    }
    if ((pgm_read_byte(&chars16[charcode-0x20][h*2+1]))>>(7-i))&0x01)
    {
        TFT9341_DrawPixel(x+i+8,y+h,color);
    }
    else
    {
        TFT9341_DrawPixel(x+i+8,y+h,phone);
    }
}
```

- [Август 2017](#)
- [Июль 2017](#)
- [Июнь 2017](#)
- [Май 2017](#)
- [Апрель 2017](#)
- [Март 2017](#)
- [Февраль 2017](#)
- [Январь 2017](#)
- [Декабрь 2016](#)
- [Ноябрь 2016](#)

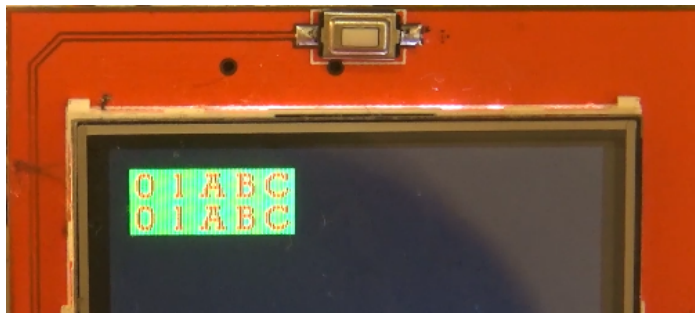


| | |
|---------|--------|
| 31 ДЕНЬ | 88 083 |
| | 9 064 |
| 07 ДНЕЙ | 22 948 |
| | 2 994 |
| 24 ЧАСА | 3 036 |
| | 713 |
| СЕГОДНЯ | 1 414 |
| | 385 |
| НАПЛИНУ | 82 |
| | 23 |

Создадим прототип нашей функции и попробуем в функции main() что-то написать на экране

```
TFT9341_FillScreen(BLACK);
TFT9341_Draw_Char(10,10,RED,GREEN,0x21,2);
TFT9341_Draw_Char(26,10,RED,GREEN,0x22,2);
TFT9341_Draw_Char(42,10,RED,GREEN,0x23,2);
TFT9341_Draw_Char(58,10,RED,GREEN,0x24,2);
TFT9341_Draw_Char(74,10,RED,GREEN,0x25,2);
TFT9341_Draw_Char(10,26,RED,GREEN,0x21,1);
TFT9341_Draw_Char(18,26,RED,GREEN,0x22,1);
TFT9341_Draw_Char(26,26,RED,GREEN,0x23,1);
TFT9341_Draw_Char(34,26,RED,GREEN,0x24,1);
TFT9341_Draw_Char(42,26,RED,GREEN,0x25,1);
_delay_ms(1000);
```

Соберём код, прошьём контроллер и посмотрим результат



Теперь попробуем написать функцию отрисовки целой строки на экране

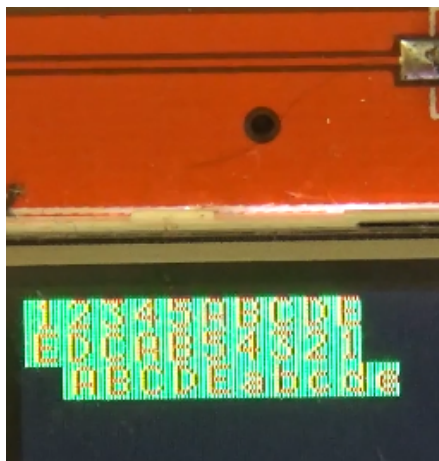
```
//-----
void TFT9341_String(unsigned int x, unsigned int y,
unsigned int color,unsigned int phone,char *str,
unsigned char size)
{
    while(*str)
    {
        if((x + (size*8)) > X_SIZE)
        {
            x = 1;
            y = y + (size*8);
        }
        TFT9341_Draw_Char(x, y, color, phone,*str, size);
        x += size*8;
        *str++;
    }
}
//-----
```

При кажущейся сложности отрисовки строки, функция оказалась не такой и тяжёлой. Постепенно сдвигая указатель на следующий символ в строке и отслеживая конец строки, мы потихоньку выводим символы слева направо. Но также не забываем отслеживать правый край дисплея, чтобы не уйти за него.

Также напишем прототип данной функции и в main() напишем тест для проверки данной функции

```
_delay_ms(1000);
TFT9341_FillScreen(BLACK);
TFT9341_String(1, 1, RED, GREEN, "12345ABCDE", 1);
TFT9341_String(1, 9, RED, GREEN, "EDCBA54321", 1);
TFT9341_String(10, 17, RED, GREEN, "ABCDEabcde", 1);
_delay_ms(1000);
TFT9341_FillScreen(BLACK);
while (1)
```

Соберём код, прошьём контроллер и посмотрим наши строки на практике



А также в бесконечный цикл мы напишем тест вывода случайных нулей и единиц размером 8 пикселей в случайные места экрана

```
TFT9341_FillScreen(BLACK);
while (1)
{
    TFT9341_Draw_Char((rand()%15)*16,
    (rand()%20)*16, GREEN, BLACK, 0x21+(rand()%2), 2);
}
}
```

Скомпилируем наш код, прошьём контроллер и посмотрим результат



Я думаю, просмотрев полностью данное занятие, вы стали гораздо грамотнее в программировании микроконтроллеров, а также немного начали разбираться в написании кода для контроллеров дисплеев с параллельным подключением.

*Предыдущая
часть*

*Программирование
МК AVR*

*Следующий
урок*

Исходный код

Техническая документация на контроллер дисплея ILI9341

Программатор и символьный дисплей LCD 20×4 можно приобрести здесь:

Программатор (продавец надёжный) [USBASP USBISP 2.0](#)

[Дисплей LCD 20×4](#)

Смотреть ВИДЕОУРОК (нажмите на картинку)



Назначение перевода Я хочу поодержать проект Narod Stream

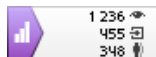
Сумма руб.

Отправить

[Главная](#) | [Новости](#) | [Уроки по программированию МК](#)
| [Программирование микроконтроллеров AVR](#)
| [Программирование микроконтроллеров STM32](#)
| [Программирование микроконтроллеров PIC](#) | [Тесты устройств](#)
и аксессуаров
| [Ссылки](#) | [Форум](#) | [Помощь](#)



Google



© 2017 Narod Stream

