

Первое включение *TFT LCD 2.4"* с драйвером *R61520* под управлением *STM32*

К.А. Кузнецов, Н.О. Кондратьев, М.В. Трубин

ФГБОУ ВО «Новосибирский государственный технический университет», Новосибирск, Россия. 630090, просп. К. Маркса, д. 20.

Аннотация: В статье описывается работа с *TFT LCD* дисплеем, на примере модели с драйвером *R61520*, под управлением отладочной платы *STM32VLDISCOVERY*. Описываются способы, позволяющие сократить время разработки при работе с дисплеем. Рассматриваются основные проблемы, возникающие при начальной работе и инициализации. Объясняется работа параллельного интерфейса 8080 и основные функции взаимодействия с ним, написанные с помощью библиотеки *SPL*. В статье даны поясняющие иллюстрации и типовые эпюры сигналов, а также таблицы состояний, что позволяет использовать ее как справочное пособие при разработки различных робототехнических устройств, использующих дисплеи описываемого типа. Приведены программные коды для выполнения важнейших операций при работе с дисплеем. Дана принципиальная схема подключения дисплея к микропроцессору. Также приведены иллюстрации, демонстрирующие графические возможности дисплея.

Ключевые слова: *TFT LCD shield*, ЖК, дисплей, *R61520*, *SPFD5408*, *STM32VLDISCOVERY*, *STM32*, *Arduino*, параллельный интерфейс 8080.

ВВЕДЕНИЕ

Статьи по робототехнике, в которых описываются решения конкретных проблем по созданию отдельных узлов, пользуются большим спросом у читателей [1–20]. На рынке электронных устройств широкое распространение получили разнообразные дисплеи. Большую часть из них представляют дешёвые китайские изделия *TFT LCD*. При своей низкой стоимости данные дисплеи имеют ряд серьезных упущений, с которыми приходится сталкиваться разработчику. При этом, несмотря на все сложности, за небольшую цену, можно приобрести дисплей с довольно широкими функциональными возможностями.

Данный материал посвящен тонкостям работы, которые помогут быстрее подключить и настроить *TFT* дисплей. Имеется множество разновидностей дисплеев. **Важнейшее отличие типов дисплеев кроется в установленном в них управляющем драйвере.** Каждый драйвер имеет отличный от других способ настройки. Поэтому основной материал направлен на пояснение начальной инициализации дисплея с драйвером *R61520*. В статье предложен в небольшом объеме программный код для запуска дисплея. Описанных функций и принципов работы с данным дисплеем достаточно, чтобы лучше разобраться в работе со схожими дисплеями, на которых установлены другие драйверы.

1. ОПИСАНИЕ *TFT LCD* ДИСПЛЕЯ

TFT LCD (Thin Film Transistor Liquid Crystal Display) – это жидкокристаллический дисплей, в котором используется активная матрица, управляемая тонкопленочными транзисторами. Дисплей представляет собой два оптических поляризатора, между которыми определённым образом располагаются жидкие кристаллы.

Расположение жидких кристаллов зависит от технологии построения матрицы. Регулирование яркости ячеек ЖК-дисплея осуществляется путём приложения внешнего электрического поля к жидким кристаллам.

Матрица жидкокристаллического дисплея представляет собой упорядоченную совокупность большого количества пикселей. Каждый пиксель, в свою очередь, состоит из трех субпикселей. Субпиксели отличаются лишь цветом выходного светофильтра (*R* – красный светофильтр, *G* – зеленый, *B* – синий). Для нормальной работы жидкокристаллических дисплеев необходима внешняя подсветка, расположенная позади ЖК-матрицы. В ней, как правило, используются, могут выступать яркие светодиоды белого цвета. Свет, проходя через слой ЖК-матрицы, создает на экране дисплея требуемое изображение, как показано на *Рис. 1*.

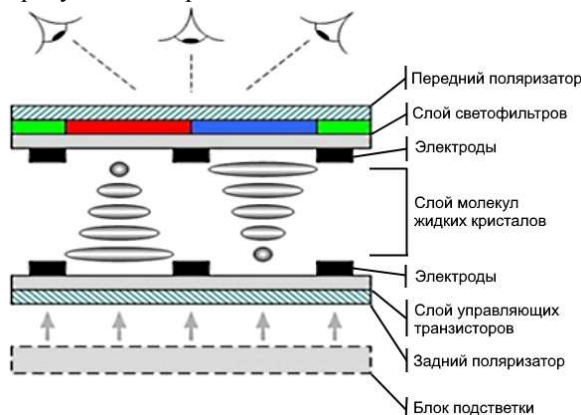


Рис. 1. Схема строения ЖК-дисплея типа *TN*

Изменение напряженности приложенного электрического поля осуществляется при помощи тонкопленочных транзисторов, установленных в каждом субпикселе ЖК-матрицы. Цвет свечения пикселя задается с

помощью светофильтров, располагающихся над пикселем. Если через каждый субпиксель пикселя свет проходит с одинаковой интенсивностью, то цвет пикселя будет белым. Изменяя интенсивность света, проходящего через субпиксель, можно регулировать цвет самого пикселя.

Ознакомившись с преимуществами и недостатками разных технологий построения матриц, можно сделать вывод, что данный дисплей выполнен по технологии *TFT TN*. Это разновидность жидких кристаллов, обладающих витой структурой. Описание технологий построения матриц дано в [15, с.5].



Рис. 2. Внешний вид *TFT LCD* дисплея спереди

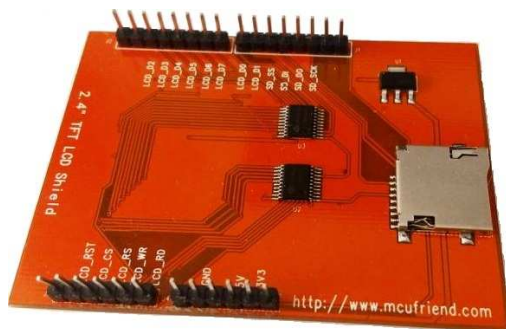


Рис. 3. Внешний вид *TFT LCD* дисплея сзади

Для драйвера *R61520* не найдено документации в открытом доступе. Выдвинуто предположение, что параметры из документации на драйвер *SPFD5408*, изначально заявленным поставщиком при продаже, идентичны. Поэтому основные параметры ниже взяты из документации на драйвер *SPFD5408* [16].

Основные характеристики дисплея:

- напряжение питания от +3,3 до +5 В;
- напряжение логической «1» от +1,65 до +3,3 В;
- ток потребления до 300 мА;
- диагональ 2,4 дюйма;
- разрешающая способность 320 × 240 пикселей с индивидуальным контролем;
- максимальное количество оттенков 262144;
- наличие параллельного 8080 интерфейса;
- сенсорная панель;
- белая подсветка;
- встроенный разъём для *microSD* карты;
- размер модуля 71 × 52 × 7 мм³.

2. ОПИСАНИЕ РАБОТЫ ПАРАЛЛЕЛЬНОГО ИНТЕРФЕЙСА 8080

Параллельный интерфейс состоит из восьми двунаправленных сигналов данных *D[7:0]*, и сигналов управления *RD#*, *WR#*, *D/C#* и *CS#*. Знак # означает инверсию сигнала. Логический «0» на *D/C#* говорит о чтении/записи команды, а логическая «1» о чтении/записи данных. При переходе 0 → 1 сигнала *RD#* происходит фиксация данных при операции чтения, когда *CS#* находится в логическом «0». При переходе 0 → 1 сигнала *WR#* происходит фиксация данных при операции записи данных/команд, когда *CS#* находится в логическом «0». На Рис. 4 показан пример эпор сигналов при операции записи, а на Рис. 5 – при операции чтения. В Табл. 1 даны состояния сигналов управления.

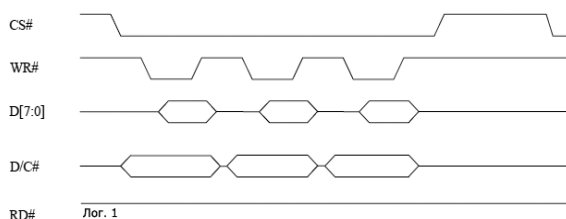


Рис. 4. Пример операции записи

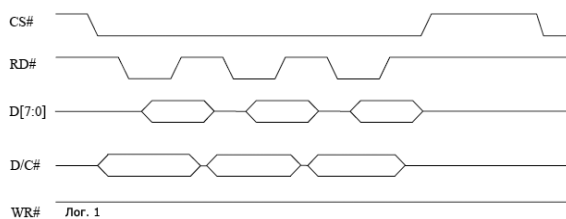


Рис. 5. Пример операции чтения

Таблица 1.

Состояние сигналов управления

Функция	RD#	WR#	CS#	D/C#
Запись команды	H	L→H	L	L
Чтения состояния	L→H	H	L	L
Запись данных	H	L→H	L	H
Чтение данных	L→H	H	L	H

Примечание. H – логическая «1»; L – логический «0»; L→H – переход сигнала из «0» в «1».

Преимущества параллельного интерфейса:

- высокая пропускная способность.

Недостатки параллельного интерфейса:

- большое количество соединительных проводов;
- полудуплексная передача данных;
- сложность синхронизации сигналов при передаче данных на высоких частотах;
- плохая помехозащищённость.

Учитывая все эти недостатки желательно делать линию передачи данных длиной не более 10 см.

3. ЭКОНОМИЯ ВРЕМЕНИ РАЗРАБОТКИ С ПОМОЩЬЮ ARDUINO

Изначально при покупке был заявлен драйвер SPFD5408. Все первоначальные эксперименты под управлением отладочной платы STM32VLDISCOVERY были неудачны, поэтому было принято решение запустить дисплей на Arduino UNO. Под эту плату и многие другие, написано большое количество библиотек, с помощью которых можно запустить дисплей и узнать, какой драйвер установлен на нём. Уточнив при покупке подобных дисплеев у поставщика, какой драйвер располагается в дисплее, можно избежать проблемы подбора различных библиотек. Это экономит время при разработке.

Используя готовую библиотеку для драйвера SPFD5408, появились первые результаты, которые говорили о том, что дисплей рабочий, но драйвер не соответствует ожидаемому типу. На дисплее вся информация дублировалась два раза в разных частях экрана, как показано на Рис. 6.



Рис. 6. TFT LCD дисплей с раздвоенным экраном

Несмотря на то, что в данных дисплеях предусмотрена возможность получения идентификатора драйвера, без сведений о том, какой тип драйвера используется, отыскать нужный его тип можно лишь перебором. Поэтому было решено осуществить поиск на зарубежных форумах, где и найдено описание подобной проблемы. Оказалось, что в дисплее применён драйвер R61520, документации на который нет в открытом доступе. На форуме предложено воспользоваться библиотекой *Mcufriend_kbv* [17]. После ее использования дисплей стал правильно функционировать, как показано на Рис. 7.

Далее, зная тип драйвера и имея рабочую библиотеку для Arduino, можно запустить дисплей на STM32. Это позволило сократить время на разработку.

Описание библиотек на Arduino для различных драйверов можно посмотреть в [18].



Рис. 7. Рабочий TFT LCD дисплей

4. ПЕРВОЕ ВКЛЮЧЕНИЕ НА STM32

Для работы с дисплеем используются шестнадцать выводов. Восемь контактов модуля предназначены для передачи данных от микроконтроллера в дисплей, либо от дисплея в микроконтроллер.

Четыре контакта задействованы для управляющих сигналов. Дополнительно к ним имеется вывод для сброса дисплея. Также имеются три вывода для «Земли», «Питание + 3,3 В» и «Питание + 5 В». Схема подключения приведена на Рис. 8.

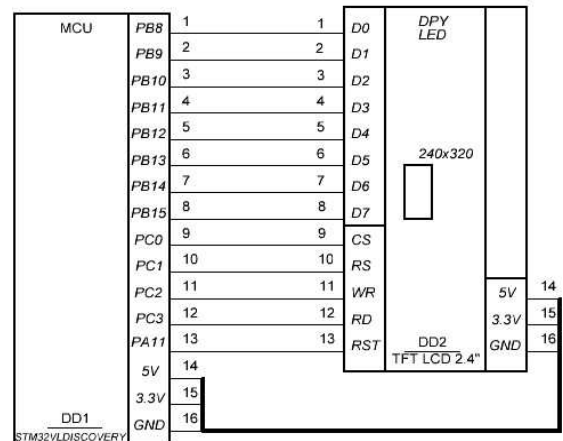


Рис. 8. Схема подключения TFT LCD

Дисплей имеет слот для SD-карты и сенсорный экран. Сенсорная часть дисплея использует 4 контакта. Для работы с microSD картой отводится 4 контакта. В некоторых публикациях пишут, что дисплей имеет возможность работы через интерфейс SPI. Но в данном дисплее выводов для этого интерфейса не найдено, кроме выводов для работы с microSD через SPI. Также в этом дисплее присутствует кнопка, которую можно использовать для пользовательских функций.

Далее следует объявление служебных команд, которые будут использоваться в дальнейшем при написании функций.


```

#define LCD_IOP GPIOB // LCD_PINS
#define LCD_CTP GPIOC // LCD_CONT
#define LCD_PINS (GPIO_Pin_8 | GPIO_Pin_9 |
GPIO_Pin_10 | GPIO_Pin_11 |
GPIO_Pin_12 | GPIO_Pin_13 | GPIO_Pin_14 |
GPIO_Pin_15)
#define LCD_CONT (GPIO_Pin_0 | GPIO_Pin_1 |
GPIO_Pin_2 | GPIO_Pin_3)
#define CS GPIO_Pin_0
#define RS GPIO_Pin_1
#define WR GPIO_Pin_2
#define RD GPIO_Pin_3
#define RST GPIO_Pin_11
// reset
#define RESET_IDLE GPIO_SetBits(GPIOA, RST);
#define RESET_ACTIVE GPIO_ResetBits(GPIOA, RST);
// chip select
#define CS_IDLE GPIO_SetBits(LCD_CTP, CS);
#define CS_ACTIVE GPIO_ResetBits(LCD_CTP, CS);
// command/data
#define RS_IDLE GPIO_SetBits(LCD_CTP, RS);
#define RS_ACTIVE GPIO_ResetBits(LCD_CTP, RS);
// read
#define RD_IDLE GPIO_SetBits(LCD_CTP, RD);
#define RD_ACTIVE GPIO_ResetBits(LCD_CTP, RD);
// write
#define WR_IDLE GPIO_SetBits(LCD_CTP, WR);
#define WR_ACTIVE GPIO_ResetBits(LCD_CTP, WR);
// write strobe
#define WR_STROBE {WR_ACTIVE; WR_IDLE;}

```

Одной из проблем, которые могут возникнуть при инициализации – это отсутствие задержек между некоторыми исполняемыми командами. Из-за этого может не работать дисплей, потому что не успевают выполняться некоторые команды. При этом может быть черный экран или экран с хаотичным рисунком. Для исключения этой ошибки стоит увеличить время задержек.

Далее описаны инициализация, основные функции и способы формирования сигналов для запуска дисплея.

```

void TFT1520_Reset(void) {
    CS_IDLE; RD_IDLE; WR_IDLE;
    RESET_IDLE;
    delay_us(5); // Задержка на 5 мкс
    RESET_ACTIVE; delay_us(15);
    RESET_IDLE; delay_us(15); }
void TFT1520_init(void) {
    LCDIO_Setup.GPIO_Pin = LCD_PINS;
    LCDIO_Setup.GPIO_Speed = GPIO_Speed_10MHz;
    LCDIO_Setup.GPIO_Mode = GPIO_Mode_Out_PP;
    GPIO_Init(LCD_IOP, &LCDIO_Setup);
    GPIO_SetBits(LCD_IOP, LCD_PINS);
    LCDCTR_Setup.GPIO_Pin = LCD_CONT | RST;
    LCDCTR_Setup.GPIO_Speed = GPIO_Speed_10MHz;
    LCDCTR_Setup.GPIO_Mode = GPIO_Mode_Out_PP;
    GPIO_Init(LCD_CTP, &LCDCTR_Setup);
    GPIO_SetBits(LCD_CTP, LCD_CONT);
    GPIO_SetBits(GPIOA, RST);
    TFT1520_Reset();
    TFT1520_SendCommand(0x01); delay_us(2500);
    TFT1520_SendCommand(0x28);
    TFT1520_SendCommand(0xB0);
    TFT1520_SendData(0x00);
    TFT1520_SendCommand(0xC0);
    TFT1520_SendData(0x0A);
    TFT1520_SendCommand(0x11); delay_us(1500);
    TFT1520_SendCommand(0x29);
    TFT1520_SendCommand(0x3A);
    TFT1520_SendData(0x55);
}

```

```

TFT1520_SetRotation(0);
TFT1520_InvertDisplay(0); }

```

Первоначально настраиваются выходы микроконтроллера. Для записи данных в дисплей выходы микроконтроллера настраиваются на выход. Но если нужно будет считывать данные с дисплея, то требуется перепрограммировать выходы линии данных $D[7:0]$ на вход.

Далее происходит сброс дисплея. В этой функции есть задержки. Без них в дисплее не успеют пройти процессы, необходимые для сброса. После этого записываются необходимые данные в управляющие регистры для настройки дисплея.

Следующая функция служит для передачи данных по параллельному интерфейсу 8080: в данной функции на выводах $PB8-PB15$ формируется не инверсная команда, состоящая, из 8 бит информации. После этого формируется импульс, сигнализирующий о записи в дисплей информации сформированной на выводах $PB8-PB15$.

```

// Формирование команды на порту
void pushData(unsigned char data) {
    LCD_IOP->ODR = (int)((int)data << 8);
    WR_STROBE; }

```

Следующие функции отправляют команды или данные. Эти функции показаны в *таблице 1*.

```

void TFT1520_SendCommand(uint16_t index) {
    CS_ACTIVE;
    RS_ACTIVE;
    pushData(index);
    CS_IDLE; }
void TFT1520_SendData(uint16_t data) {
    CS_ACTIVE;
    RS_IDLE;
    pushData(data);
    CS_IDLE; }

```

Также на этапе настройки есть следующая функция:

```

void TFT1520_InvertDisplay(uint8_t i) {
    TFT1520_SendCommand(i ? 0x21 : 0x20); }

```

Эта функция предназначена для аппаратного инвертирования пикселей дисплея. Если в неё передать 1, то все чёрные пиксели поменяют свой цвет на белый, а белые – на черный.

Также немаловажной функцией будет установка ориентации дисплея:

```

uint16_t X_SIZE = 0;
uint16_t Y_SIZE = 0;
void TFT1520_SetRotation(uint8_t r) {
    TFT1520_SendCommand(0x36);
    switch(r) {
        case 0:
            TFT1520_SendData(0x48);
            X_SIZE = 240; Y_SIZE = 320;
            break;
        case 1:
            TFT1520_SendData(0x28);
            X_SIZE = 320; Y_SIZE = 240;
            break;
        case 2:

```

```
TFT1520_SendData(0x88);
X_SIZE = 240; Y_SIZE = 320;
break;
case 3:
TFT1520_SendData(0xE8);
X_SIZE = 320; Y_SIZE = 240;
break; } }
```

Для настройки ориентации дисплея необходимо вызвать данную функцию со значением от 0 до 3. Изменяя регистры, отвечающие за ориентацию, изменяется положение начальной точки на экране, т.е. координаты (0,0). Положений начальной точки всего лишь 4, в каждом из углов экрана.

Следующим шагом будет заполнение дисплея одним цветом (константой). Для этого понадобится функция установки курсора на требуемые пиксель и функция заполнения пикселя определённым цветом.

```
void TFT1520_SetAddrWindow(uint16_t x, uint16_t
y, uint16_t x1, uint16_t y1) {
TFT1520_SendCommand(0x2A);
TFT1520_SendData(x >> 8);
TFT1520_SendData(x);
TFT1520_SendData(x1 >> 8);
TFT1520_SendData(x1);
TFT1520_SendCommand(0x2B);
TFT1520_SendData(y >> 8);
TFT1520_SendData(y);
TFT1520_SendData(y1 >> 8);
TFT1520_SendData(y1); }
void TFT1520_DrawPixel(uint16_t x, uint16_t y,
uint16_t color) {
if((x<0)|| (y<0)|| (x>=X_SIZE)|| (y>=Y_SIZE))
return;
TFT1520_SetAddrWindow(x, y, x, y);
TFT1520_SendCommand(0x2C);
TFT1520_SendData(color >> 8);
TFT1520_SendData(color & 0xFF); }
```

Далее приведён небольшой пример, заполняющий попиксельно дисплей сначала белым цветом, потом чёрным.

```
int main(void) {
uint16_t x = 0; uint16_t y = 0;
TFT1520_init(); delay_us(1000000);
TFT1520_SetRotation(1);
while(1) {
for(x = 0; x < 320; x++) {
for(y = 0; y < 240; y++) {
// заполнение дисплея белым попиксельно
TFT1520_DrawPixel(x, y, 0xFFFF);
}
}
for(x = 0; x < 320; x++) {
for(y = 0; y < 240; y++) {
// заполнение дисплея чёрным попиксельно
TFT1520_DrawPixel(x, y, 0x0000);
}}}}
```

Изложенных выше функций вполне достаточно для настройки и проверки работоспособности дисплея. Пример вывода текстовой и графической информации можно увидеть на Рис. 9-10.

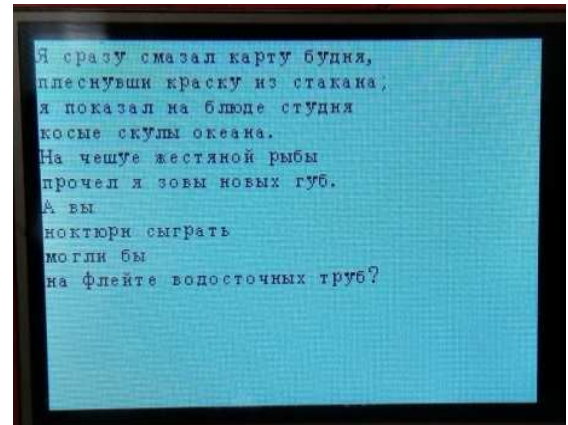


Рис. 9. Возможность вывода текстовой информации под управлением STM32

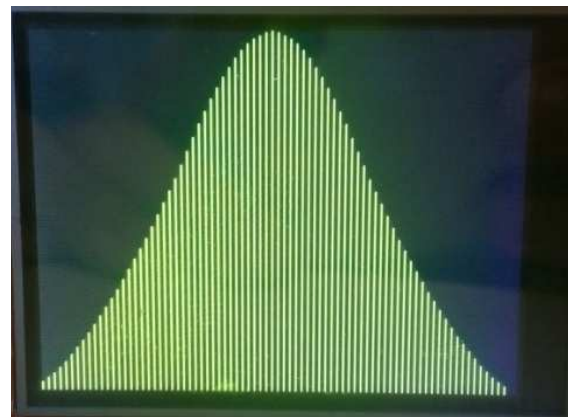


Рис. 10. Возможность вывода графической информации под управлением STM32

ВЫВОДЫ

Данный дисплей кажется довольно сложным, особенно при первом знакомстве. Однако это компенсируется низкой ценой и разнообразными функциональными возможностями в виде сенсорной панели и слота под SD-карту, присутствующим в дисплее. Данный функционал позволяет расширить область применения подобных дисплеев, что является весомым плюсом.

TFT LCD не обладает большой яркостью, по сравнению с LED или OLED индикациями. Однако имеющейся яркости вполне достаточно для решения многих задач. Даже под большим углом обзора картинка с дисплея хорошо видна. Ещё одним преимуществом будет огромное количество цветных оттенков, более 262 тысяч возможных, для одного пикселя.

ЛИТЕРАТУРА

[1] Печников А.Л., Жмудь В.А., Трубин В.Г., А.Б. Колкер. Перспективы развития робототехнических учебных стендов для высшего специального образования в области робототехники, автоматизации и мехатроники. Информатика, вычислительная техника и автоматизация. 2012. №2. Т.5. С. 89–95

- [2] Печников А.Л., Жмудь В.А., Трубин В.Г. Удаленное управление роботом посредством хттрр-протокола. Сборник научных трудов НГТУ. 2013. № 3(73). С. 85 – 92.
- [3] Ескин А.В., Жмудь В.А., Трубин В.Г. Реализация дистанционного управления по радиоканалу Bluetooth платформой, моделирующей работу роботизированных средств. Автоматика и программная инженерия. 2013. № 1 (3). С. 82–87.
- [4] Ескин А.В., Жмудь В.А., Трубин В.Г. Построение платформы, моделирующей работу роботизированных средств на базе конструктора Lego Mindstorms NXT 2.0 в части управления электродвигателями. Автоматика и программная инженерия. 2013. № 1 (3). С. 88–94.
- [5] А.В. Ескин, В.А. Жмудь, В.Г. Трубин. Беспроводной удлинитель последовательного порта на базе радиоканала Bluetooth. Автоматика и программная инженерия. 2013. № 2 (4). С. 42–47.
- [6] А.В. Ескин, В.А. Жмудь, В.Г. Трубин. STM32VLDISCOVERY – средство для быстрой разработки опытных образцов цифровых систем управления. Автоматика и программная инженерия. 2013. № 3 (5). С. 32–39.
- [7] А.В. Ескин, В.А. Жмудь, В.Г. Трубин. Экономичная реализация графического интерфейса пользователя на базе одноплатного компьютера Raspberry Pi. Автоматика и программная инженерия. 2014. № 2 (8). С. 22–38.
- [8] В.Г. Трубин, А. В. Ескин, А.Л. Печников, В.А. Жмудь. Использование сети интернет для интерактивной лабораторной работы с дистанционным управлением моделью робота снегоборщика. Автоматика и программная инженерия. 2014. 1(7). С. 95–103.
- [9] Жмудь В.А., Трубин И.В., Трубин М.В. Обмен данными между компьютером и микроконтроллером STM32F100 по последовательному интерфейсу связи RS-232. Автоматика и программная инженерия. 2015. № 1 (11). С. 45–51.
- [10] Жмудь В.А., Трубин И.В., Трубин М.В. Проектирование сенсорных кнопок на базе микросхемы TTP-224. Автоматика и программная инженерия. 2015. № 1 (11). С. 70–74.
- [11] Федоров Д.С., Ивойлов А.Ю., Жмудь В.А., Трубин В.Г. Использование измерительной системы MPU 6050 для определения угловых скоростей и линейных ускорений. Автоматика и программная инженерия. 2015. № 1 (11). С. 75–80.
- [12] Д.С. Федоров, А.Ю. Ивойлов, В.А. Жмудь, В.Г. Трубин. Разработка системы стабилизации угла отклонения балансирующего робота. Автоматика и программная инженерия. 2015. № 2 (12). С. 16–34.
- [13] В.А. Жмудь, А.С. Каменская, К.В. Курбетьев, В.Г. Трубин. Графический OLED дисплей UG-2864ASGGG14: первое включение. Автоматика и программная инженерия. 2016. № 1 (15). С. 29–37.
- [14] STM32VLDISCOVERY – платформа для построения простой системы сбора данных. Лабораторная работа: учеб.-метод. пособие / В. А. Жмудь, В. Г. Трубин, А. В. Ескин, А. Л. Печников. - Новосибирск: Издательство НГТУ, 2014. - 174 с. - 50 экз. - ISBN 978-5-7782-2599-2.
- [15] Гонин М. Kyocera Display: применение TFT-дисплеев в промышленности и медицине. Новости электроники: информационно-технический журнал. 2014. № 2. URL: http://www.compel.ru/wordpress/wp-content/uploads/2014/03/NE_02_2014.pdf.
- [16] Документация на драйвер SPFD5408: 720-channel 6-bit Source Driver with System-on-chip for Color Amorphous TFT LCDs: <http://www.rhydolabz.com/documents/SPFD5408A.pdf>.
- [17] MCFRIEND_kbv Library for Uno 2.4, 2.8, 3.5, 3.6, 3.95 inch mcfriend Shields: форум Arduino: URL: <https://forum.arduino.cc/index.php?topic=366304.0>.
- [18] Дисплей 2.4 TFT 240x320 сенсорный + MicroSD. URL: <https://arduino-kit.ru/catalog/id/display-2.4-tft-240h320-sensornyiy-%2B-microsd>.
- [19] Документация на микроконтроллеры: STM32F100x4 STM32F100x6 STM32F100x8 STM32F100xB. STMicroelectronics: URL: <http://www.st.com/content/ccc/resource/technical/document/datasheet/dd/87/fd/2a/fb/3f/48/5c/CD00251732.pdf/files/CD00251732.pdf/jcr:content/translations/en.CD00251732.pdf>.
- [20] Документация на отладочную плату STM32VLDISCOVERY: UM0919 User Manual. STMicroelectronics: URL: http://www.st.com/content/ccc/resource/technical/document/user_manual/f3/16/fb/63/d6/3d/45/aa/CD00267113.pdf/files/CD00267113.pdf/jcr:content/translations/en.CD00267113.pdf.



Кирилл Андреевич Кузнецов
- магистрант группы ААМ-17
кафедры Автоматики НГТУ.
E-mail: kuznetsov@ngs.ru

630073, Новосибирск,
просп. К.Маркса, д. 20



Никита Олегович Кондратьев
- магистрант группы ААМ-17
кафедры Автоматики НГТУ.
E-mail: nikitok.08.95@mail.ru

630073, Новосибирск,
просп. К.Маркса, д. 20



Максим Витальевич Трубин –
бакалавр группы АА-46
кафедры Автоматики НГТУ.
E-mail: morkai@bk.ru

630073, Новосибирск,
просп. К.Маркса, д. 20

Статья поступила в редакцию 22 декабря 2018 г.

The First Insertion of TFT LCD 2.4" with Driver R61520, Controlled by STM32

K.A. Kuznetsov, N.O. Kondratyev, M.V. Trubin

Novosibirsk State Technical University, Novosibirsk, Russia, str. Prospekt K. Marksa, h.20

Abstract: This paper deals with the use of the TFT LCD display and model with R61520 driver as it's example. TFT LCD is controlled by the debug board STM32VLDISCOVERY. The article describes the ways which can help to reduce development time while working with the display. The main problems arising during initial work and initialization are examined. The work of the parallel interface 8080 and the basic functions, written using the SPL library, of interaction with it, is spoken in detail. The paper provides explanatory illustrations and typical diagrams of signals, as well as tables of states, which allows using it as a reference tool for the development of various robotic devices using displays of the type described. The program codes for performing the most important operations when working with the display are given. A schematic diagram of connecting the display to a microprocessor is given. In addition the paper gives illustrations of practical graphics capabilities of the display.

Key words: TFT LCD shield, LCD, display, R61520, SPFD5408, STM32VLDISCOVERY, STM32, Arduino, parallel interface 8080.

REFERENCES

- [1] Pechnikov A.L., Zhmud V.A., Trubin V.G., A.B. Kolkner. Perspektivy razvitiya robototekhnicheskikh uchebnykh stendov dlja vysshego special'nogo obrazovaniya v oblasti robototekhniki, avtomatiki i mehatroniki. Informatika, vychislitel'naja tehnika i avtomatizacija. 2012. №2. T.5. S. 89–95
- [2] Pechnikov A.L., Zhmud V.A., Trubin V.G. Udalennoe upravlenie robotom posredstvom xmpp-protokola. Sbornik nauchnykh trudov NGTU. 2013. № 3(73). S. 85 – 92.
- [3] Eskin A.V., Zhmud V.A., Trubin V.G. Realizacija distancionnogo upravleniya po radiokanalu Bluetooth platformoj, modelirujushhej rabotu robotizirovannyh sredstv. Avtomatika i programmnaja inzhenerija. 2013. № 1 (3). S. 82–87.
- [4] Eskin A.V., Zhmud V.A., Trubin V.G. Postroenie platformy modelirujushhej rabotu robotizirovannyh sredstv na baze konstruktora Lego Mindstorms NXT 2.0 v chasti upravleniya jelektroprivodami. Avtomatika i programmnaja inzhenerija. 2013. № 1 (3). S. 88–94.
- [5] A.V. Eskin, V.A. Zhmud, V.G. Trubin. Besprovodnoj udalitel' posledovatel'nogo porta na baze radiokanala Bluetooth. Avtomatika i programmnaja inzhenerija. 2013. № 2 (4). S. 42–47.
- [6] A.V. Eskin, V.A. Zhmud, V.G. Trubin. STM32VLDISCOVERY – sredstvo dlja bystroj razrabotki opytnyh obrazcov cifrovych sistem upravleniya. Avtomatika i programmnaja inzhenerija. 2013. № 3 (5). S. 32–39.
- [7] A.V. Eskin, V.A. Zhmud, V.G. Trubin. Jekonomichnaja realizacija graficheskogo interfejsa pol'zovatelja na baze odnoplattynogo komp'yutera Raspberry Pi. Avtomatika i programmnaja inzhenerija. 2014. № 2 (8). S. 22–38.
- [8] V.G. Trubin, A. V. Eskin, A.L. Pechnikov, V.A. Zhmud. Ispol'zovanie seti internet dlja interaktivnoj laboratornoj raboty s distancionnym upravleniem model'ju robota snegouborshhika. Avtomatika i programmnaja inzhenerija. 2014. 1(7). S. 95–103.
- [9] Zhmud V.A., Trubin I.V., Trubin M.V. Obmen dannymi mezhdju komp'yuterom i mikrokontrollerom STM32F100 po posledovatel'nomu interfejsu svjazi RS-232. Avtomatika i programmnaja inzhenerija. 2015. № 1 (11). S. 45–51.
- [10] Zhmud V.A., Trubin I.V., Trubin M.V. Proektirovanie sensoryh knopok na baze mikroshemy TTP-224. Avtomatika i programmnaja inzhenerija. 2015. № 1 (11). S. 70–74.
- [11] Fedorov D.S., Ivojlav A.Ju., Zhmud V.A., Trubin V.G. Ispol'zovanie izmeritel'noj sistemy MPU 6050 dlja opredeleniya uglovych skorostej i linejnyh uskorenij. Avtomatika i programmnaja inzhenerija. 2015. № 1 (11). S. 75–80.
- [12] D.S. Fedorov, A.Ju. Ivojlav, V.A. Zhmud, V.G. Trubin. Razrabotka sistemy stabilizacii ugla otklonenija balansirujushhego robota. Avtomatika i programmnaja inzhenerija. 2015. № 2 (12). S. 16–34.
- [13] V.A. Zhmud', A.S. Kamenskaja, K.V. Kurbet'ev, V.G. Trubin. Graficheskij OLED displej UG-2864ASGGG14: pervoe vključenje. Avtomatika i programmnaja inzhenerija. 2016. № 1 (15). S. 29–37.
- [14] STM32VLDISCOVERY – platforma dlja postroenija prostoj sistemy sbora dannyh. Laboratornaja rabota : ucheb.-metod. posobie / V. A. Zhmud, V. G. Trubin, A. V. Eskin, A. L. Pechnikov. - Novosibirsk: Izdatel'stvo NGTU, 2014. - 174 s. - 50 jekz. - ISBN 978-5-7782-2599-2.
- [15] Gonin M. Kyocera Display: primenenie TFT-displeev v promyshlennosti i medicine. Novosti jelektroniki: informacionno-tehnicheskij zhurnal. 2014. № 2. URL: http://www.compel.ru/wordpress/wp-content/uploads/2014/03/NE_02_2014.pdf.
- [16] Dokumentacija na drajver SPFD5408: 720-channel 6-bit Source Driver with System-on-chip for Color Amorphous TFT LCDs: <http://www.rhydolabz.com/documents/SPFD5408A.pdf>.
- [17] MCFRIEND_kbv Library for Uno 2.4, 2.8, 3.5, 3.6, 3.95 inch mcufriend Shields: forum Arduino: URL: <https://forum.arduino.cc/index.php?topic=366304.0>.
- [18] Displej 2.4 TFT 240h320 sensoryj + MicroSD. URL: <https://arduino-kit.ru/catalog/id/display-2.4-tft-240h320-sensornyj-%2B-microsd>.
- [19] Dokumentacija na mikrokontrollery: STM32F100x4 STM32F100x6 STM32F100x8 STM32F100xB. STMicroelectronics: URL: <http://www.st.com/content/ccc/resource/technical/document/datasheet/dd/87/fd/2a/fb/3f/48/5c/CD00251732.pdf/files/CD00251732.pdf/jcr:content/translations/en.CD00251732.pdf>.
- [20] Dokumentacija na otladochnuju platu STM32VLDISCOVERY: UM0919 User Manual. STMicroelectronics: URL: http://www.st.com/content/ccc/resource/technical/document/user_manual/f3/16/fb/63/d6/3d/45/aa/CD00267113.pdf/files/CD00267113.pdf/jcr:content/translations/en.CD00267113.pdf.



Kirill Andreevich Kuznetsov is a student of Automatics Department of NSTU.

E-mail: kuznetsovk@ngs.ru

630073, Novosibirsk,
str. Prosp. K. Marksa, h. 20



Nikita Olegovich Kondratiev is a student of Automatics Department of NSTU.

E-mail: nikitok.08.95@mail.ru

630073, Novosibirsk,
str. Prosp. K. Marksa, h. 20



Maxim Vitalievich Trubin is a student of Automatics Department of NSTU.

E-mail: morkai@bk.ru

630073, Novosibirsk,
str. Prosp. K. Marksa, h. 20

Paper was received 22 December 2018 г.