

# Внутрисхемный программатор ST-LINK/V2 (для МК серии STM8, STM32)



---

Руководство по эксплуатации

# Содержание

1. Введение.....	4
2. Начало работы .....	4
2.1. Системные требования.....	4
2.2. Требования к оборудованию.....	4
2.3. Установка утилиты STM32 ST-LINK.....	4
2.4. Деинсталляция утилиты STM32 ST-LINK .....	4
3. Пользовательский интерфейс STM32 ST-LINK Utility.....	5
3.1. Основное окно .....	5
3.2. Панель меню (Menu bar) .....	5
4. Возможности утилиты STM32 ST-LINK.....	9
4.1. Информация об устройстве.....	9
4.2. Отображение и модификация памяти .....	9
4.3. Очистка флэш-памяти .....	10
4.4. Программирование устройств.....	11
4.5. Конфигурация опционных байтов .....	12
4.6. Основные функции микроконтроллера (MCU) .....	14
4.7. Функции автоматического режима .....	15
5. Интерфейс командной строки (CLI) STM32 ST-LINK Utility.....	16
5.1. Использование командной строки .....	16
5.1.1 Команды подключения и работы с памятью.....	16
5.1.2 Основные команды .....	16
5.1.3 Флэш-команды .....	17
5.1.4 Команды опционных байтов .....	18
5.1.5 Коды возврата ST-LINK_CLI .....	21
6. Эксплуатация .....	21
6.1 Конфигурация .....	22
6.2 Подключение к приложениям STM8 .....	23
6.2.1 Стандартное соединение ERNI с плоской лентой SWIM .....	23
6.2.2 LOW COST соединение SWIM .....	24
6.2.3 Сигналы SWIM соединения для ST-LINK/V2 .....	24
6.3 Подключение к приложениям STM32 .....	26
6.4 Светодиодные индикаторы состояния ST-LINK/V2 .....	27
7. Конфигурация программного обеспечения .....	28
7.1 Обновление прошивки ST-LINK/V2.....	28
7.2 Разработка приложений STM8 .....	28



## 1. Введение

ST-LINK / V2 является внутрисхемным отладчиком / программатором для семейства микроконтроллеров STM8 и STM32 в средах разработки с помощью инструментов ST-LINK и ST-LINK/V2

Однопроводный интерфейсный модуль (SWIM) и JTAG / SWD интерфейсы облегчают обмен данными с любыми STM8 или STM32 микроконтроллерами, расположенными на плате устройства.

## 2. Начало работы

В данном разделе описаны требования и процедуры установки программного обеспечения STM32 ST-LINK Utility

### 2.1. Системные требования

Для настройки STM32 ST-LINK Utility на ПК требуется как минимум:

- ПК с USB-портом и процессором Intel® Pentium® под управлением 32-разрядной версии одной из следующих операционных систем Microsoft®:
  - Windows® XP
  - Windows 7
- 32 Мбайт оперативной памяти
- 10 Мбайт свободного места на жестком диске

### 2.2. Требования к оборудованию

Утилита STM32 ST-LINK предназначена для работы с:

- STM32 F0, STM32 F1, STM32 F2, STM32 F3, STM32 F4, STM32 L1 и серией STM32W
- ST-LINK или ST-LINK/V2 или ST-LINK/V2-ISOL внутрисхемным отладчиком/программатором

*Примечание:* В данном документе под ST-LINK/V2 подразумеваются ST-LINK/V2 и ST-LINK/V2-ISOL, которые функционально эквивалентны.

### 2.3. Установка утилиты STM32 ST-LINK

Для установки программы STM32 ST-LINK Utility выполните следующие действия и следуйте инструкциям на экране.

1. Загрузите сжатое программное обеспечение STM32 ST-LINK Utility с сайта компании ST.
2. Распакуйте содержимое .zip-файла во временный каталог.
3. Дважды щелкните на извлеченном файле setup.exe, чтобы инициировать установку, и следуйте экранным подсказкам для установки STM32 ST-LINK Utility в среду разработки. Документация по утилите находится в подкаталоге \Docs, в котором установлена STM32 ST-LINK Utility.

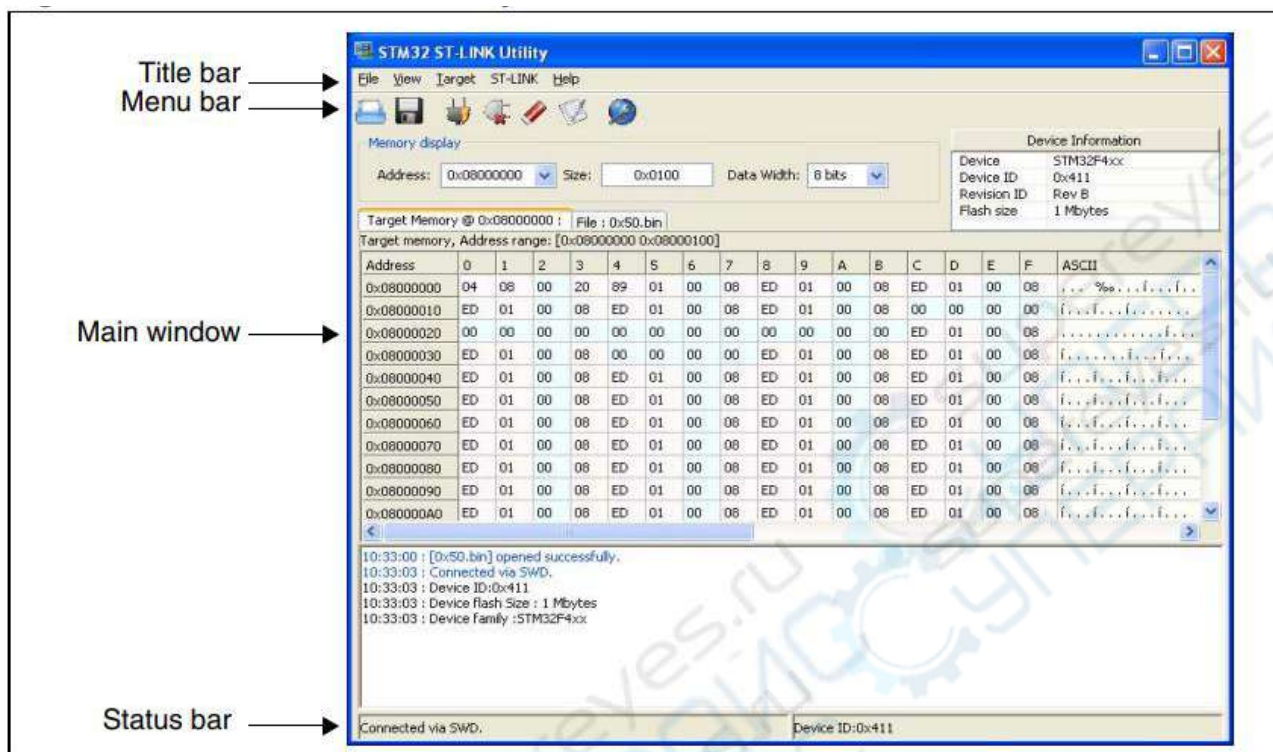
### 2.4. Деинсталляция утилиты STM32 ST-LINK

Для удаления программы STM32 ST-LINK Utility выполните следующие действия.

1. Выберите Пуск | Настройки | Панель управления.
2. Дважды щелкните на пункте Добавить или удалить программы.
3. Выберите STM32 ST-LINK Utility.
4. Щелкните на кнопке Remove (Удалить).

### 3. Пользовательский интерфейс STM32 ST-LINK Utility

#### 3.1. Основное окно



Главное окно состоит из трех зон и трех полос, как показано на рис. 1:

- Зона отображения памяти
- Зона информации об устройстве
- Зона содержимого памяти
- Строка заголовка: Название текущего меню
- Строка меню: Используйте строку меню для доступа к следующим функциям STM32 ST-LINK

Utility:

- Меню «Файл» (– File menu)
- Меню просмотра (– View menu)
- Целевое меню (Target menu)
- Меню справки (Help menu)
- Окно состояния: В строке состояния отображаются:
  - Состояние соединения и интерфейс отладки
  - Идентификатор устройства

Пользовательский интерфейс STM32 ST-LINK Utility также предоставляет дополнительные формы и описательные всплывающие сообщения об ошибках.

#### 3.2. Панель меню (Menu bar)

Панель меню позволяет пользователям выполнять функции программного обеспечения STM32 ST-LINK Utility

## 1. Меню «Файл» (– File menu)



Open file...	Открывает двоичную, Intel Hex (16-ную) или Motorola S-запись Intel.
Save file as...	Запись содержимого панели памяти в двоичном виде, в формате Intel Hex или Motorola S-record.
Close File	Закрывает загруженный файл.
Compare two files	Сравнивает два двоичных, шестнадцатеричных или srec-файла. Разница окрашивается в красный цвет на панели файлов. Если файл содержит секцию с диапазоном адресов, недоступным в другом файле, то эта секция будет окрашена в фиолетовый цвет.
Exit	Заккрытие программы STM32 ST-LINK Utility

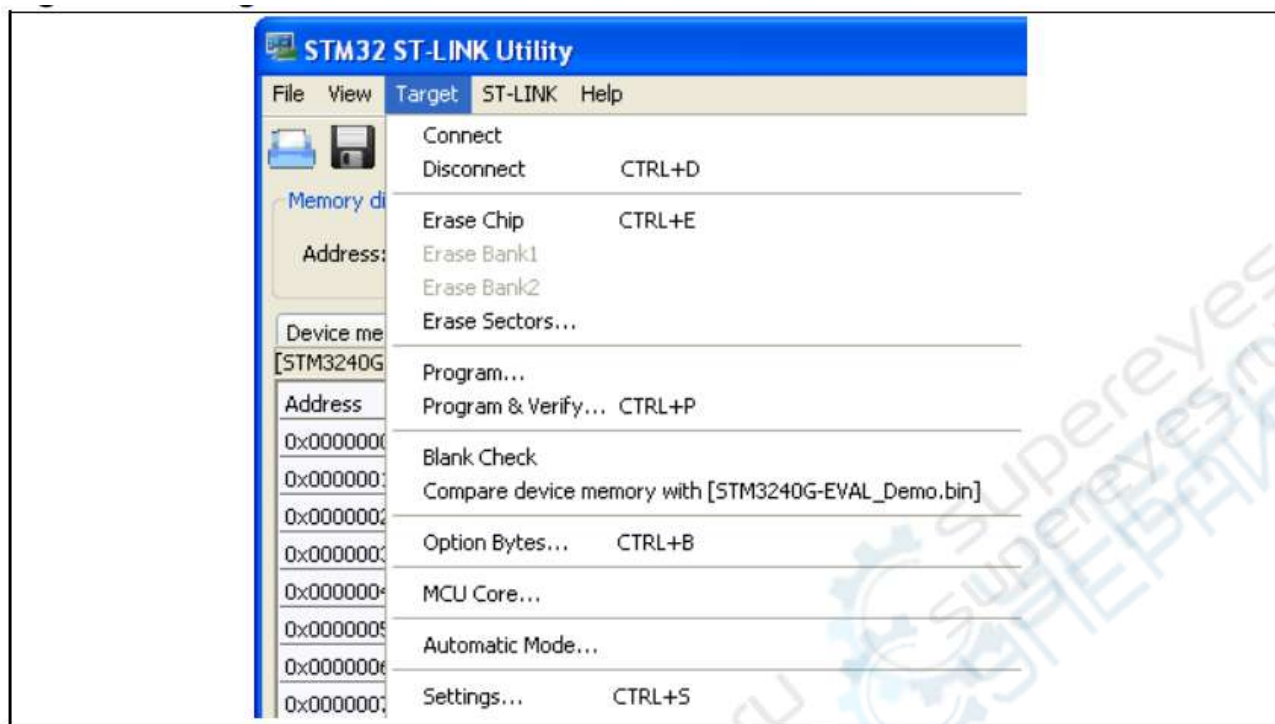
## 2. Меню просмотра (– View menu)



Binary file	Отображает содержимое загруженного двоичного файла.
Device memory	Отображает содержимое памяти устройства.



### 3. Целевое меню (Target menu)



Connect	Подключение к целевому устройству и отображение в информационной зоне устройства типа устройства, идентификатора устройства и объема флэш-памяти.
Disconnect	Отключение от целевого устройства.
Erase Chip	Выполняет массовое стирание Flash-памяти и затем отображает содержимое Flash-памяти на панели памяти.
Erase Bank1	Стирает банк 1 флэш-памяти. Это меню включено только при подключении к устройству с плотностью XL
Erase Bank2	Стирает банк 2 флэш-памяти. Данное меню включается только при подключенном к устройству с плотностью XL.
Erase Sectors...	Выбор сектора (секторов) для стирания с помощью диалогового окна "Стирание секторов" (подробнее см. раздел: Стирание флэш-памяти).
Program...	Загружает двоичный файл, файл Intel Hex или файл Motorola S-record в память устройства (Flash или RAM). Для этого выберите двоичный файл, Intel Hex или Motorola S-record, введите начальный адрес (куда поместить файл в устройстве) в диалоговом окне программы, а затем нажмите кнопку program (см. раздел: Программирование устройства).
Program & Verify...	Загружает двоичный файл, файл Intel Hex или файл Motorola S-record в память устройства (флэш-память или ОЗУ), затем выполняет проверку запрограммированных данных.
Blank Check	Проверяет, что флэш-память STM32 пуста. Если флэш-память не пуста, первый адрес с данными высвечивается в сообщении-подсказке.
Compare device memory with file	Сравнивает содержимое памяти устройства MCU с двоичным, шестнадцатеричным или srec-файлом. Разница выделена красным цветом на панели "Файл".
Option Bytes...	Открывает диалоговое окно «Option Bytes» (опционные байты)
MCU Core...	Открывает диалоговое окно функций микроконтроллера
Automatic Mode...	Открывает диалоговое окно автоматического режима (

Settings...	<p>Открывает диалоговое окно настроек для выбора интерфейса отладки (JTAG или SWD).</p> <p>Опция “подключиться при сбросе” (connect under reset) позволяет вам подключиться к целевому объекту перед выполнением какой-либо инструкции.</p> <p>Это полезно во многих случаях, например, когда целевой объект содержит код, отключающий контакты JTAG/SWD.</p> <p>При подключении к целевому устройству через ST-LINK/V2 в поле со списком «Напряжение питания» отображается целевое напряжение.</p> <p>При подключении к устройству STM32 F2 или STM32 F4 с помощью ST-LINK комбинация «Напряжение питания» позволяет выбрать напряжение питания вашего целевого устройства, чтобы иметь возможность правильно запрограммировать флэш-память</p>
-------------	--

*Примечание:* Опция "connect under reset" доступна только для ST-LINK/V2 и в режиме SWD.

Для режима JTAG опция " connect under reset" доступна начиная с версии прошивки ST-LINK/V2 V2J15Sx.

Контакт RESET разъема JTAG (вывод 15) должен быть подключен к контакту сброса устройства.

#### 4. Меню ST-LINK



Firmware update	<p>Отображает версию прошивки ST-LINK и ST-LINK/V2 микропрограммного обеспечения и обновляет его до последней версии:</p>
	<p>ST-LINK: V1J12S3</p> <p>ST-LINK/V2: V2J16S3</p>

#### 5. Меню справки (Help menu)



STM32 ST-LINK Utility User Manual	Открывается руководство пользователя утилиты STM32 ST-LINK Utility.
ST-LINK User Manual	Открывается руководство пользователя ST-LINK.
ST-LINK/V2 User Manual	Открывается руководство пользователя ST-LINK/V2.
About...	Отображает версию программного обеспечения STM32 ST-LINK Utility и информацию об авторских правах.



## 4. Возможности утилиты STM32 ST-LINK

В этом разделе представлено подробное описание использования функций STM32 ST-LINK Utility:

- Информация об устройстве.
- Отображение и изменение памяти.
- Очистка флэш-памяти.
- Программирование устройства.
- Конфигурация дополнительных байтов
- Основные функции микроконтроллера
- Функции автоматического режима

### 4.1. Информация об устройстве

В зоне "Информация об устройстве" отображается информация, как показано на рисунке:

Device Information	
Device	STM32F10xxx High-density device
Device ID	0x414
Revision ID	Rev Z
Flash size	512 Kbyte

Device:	Семейство подключенных устройств STM32. Каждый тип устройства включает в себя множество устройств с различными характеристиками, такими как объем флэш-памяти, объем оперативной памяти и периферийные устройства.
Device ID:	Идентификационный код устройства MCU, расположенный на внешней карте памяти PPB
Revision ID:	Идентификатор версии подключенного устройства MCU
Flash size:	Размер встроенной флэш-памяти

### 4.2. Отображение и модификация памяти

Помимо зоны информации об устройстве, главное окно содержит еще 2 зоны:

- Отображение памяти
- Данные памяти

**Отображение памяти:** Эта зона содержит три поля редактирования:

Address:	Начальный адрес памяти, из которой производится чтение.
Size:	Объем считываемых данных.
Data width:	Ширина отображаемых данных (8-битная, 16-битная или 32-битная).

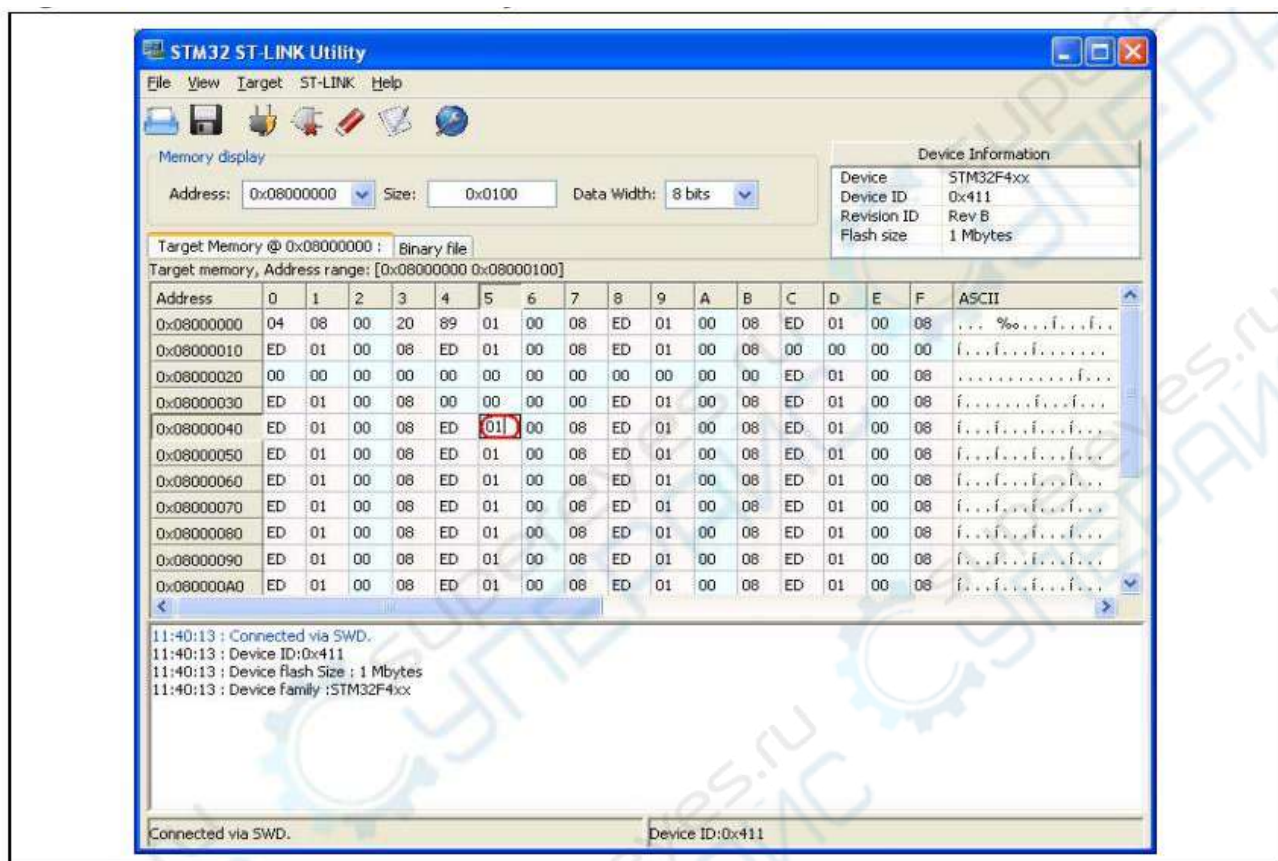
**Данные памяти:** В этой зоне отображаются данные, считанные из файла или содержимое памяти подключенного устройства. Перед загрузкой содержимое файла можно изменить.

Чтобы использовать эту зону для отображения содержимого двоичного файла, файла Intel Hex или файла Motorola S-record, перейдите в меню Файл | Открыть файл...(File | Open file...)

Чтобы использовать эту зону для считывания и отображения содержимого памяти подключенного устройства, введите начальный адрес памяти, размер данных и ширину данных в зоне отображения памяти, а затем нажмите Enter.

После считывания данных вы также можете изменить каждое значение, просто дважды щелкнув по соответствующей ячейке, как показано на рисунке 8. Вы также можете сохранить содержимое

памяти устройства в двоичный файл, Intel Hex или Motorola S-record, используя меню Файл | Сохранить файл как...(File | Save file as...)



*Примечание:* Когда в зоне данных памяти отображается содержимое памяти устройства, любые изменения автоматически применяются к чипу. Вы можете изменять пользовательскую флэш-память, оперативную память и регистры периферийных устройств.

Для устройств STM32F2 и STM32 F4 вы можете изменить область ОТР непосредственно из зоны данных памяти.

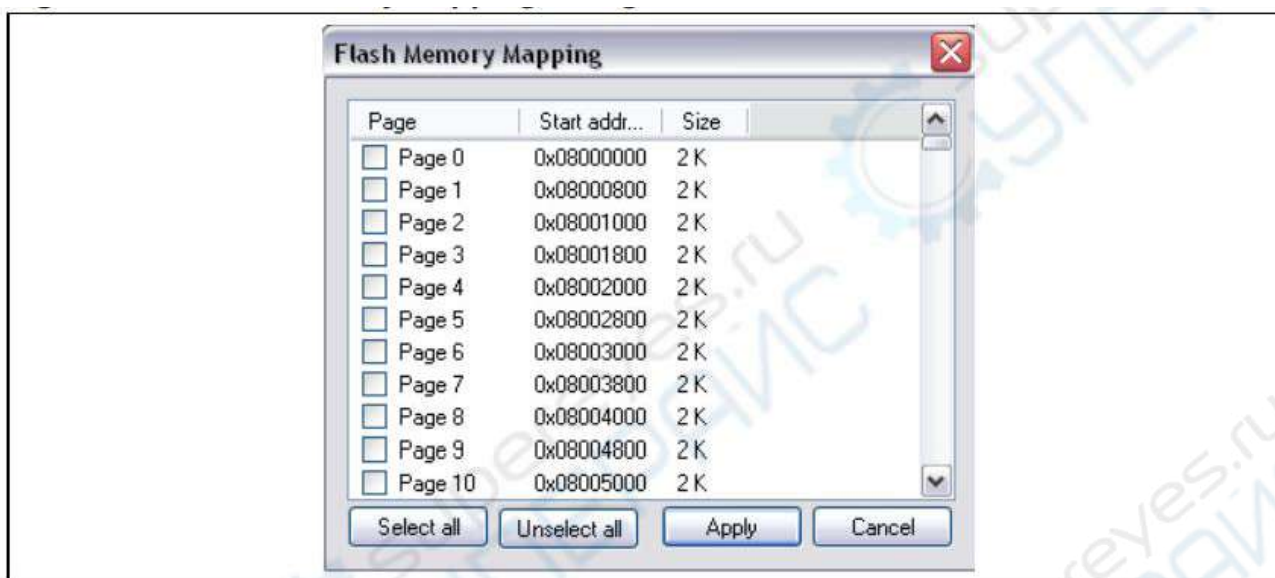
### 4.3. Очистка флэш-памяти

Существует два типа очистки флэш-памяти:

- Массовая очистка флэш-памяти: стирание всех секторов флэш-памяти подключенного устройства. Это делается щелчком по меню Target | Erase Chip.

- Очистка сектора флэш-памяти: Стирание выбранных секторов флэш-памяти. Чтобы выбрать сектор(ы), перейдите в раздел Target | Erase Sectors..., который затем отображает диалоговое окно сопоставления флэш-памяти, в котором вы выбираете сектор (сектора) для удаления, как показано на рисунке.

- Кнопка "Выбрать все" (Select all) позволяет выбрать все страницы флэш-памяти.
- Кнопка Отменить выбор всех (Deselect all) отменяет выбор всей выбранной страницы.
- Кнопка "Отмена" (Cancel) отменяет операцию очистки, даже если выбраны некоторые страницы.
- Кнопка «применить» (Apply) удаляет все выбранные страницы.

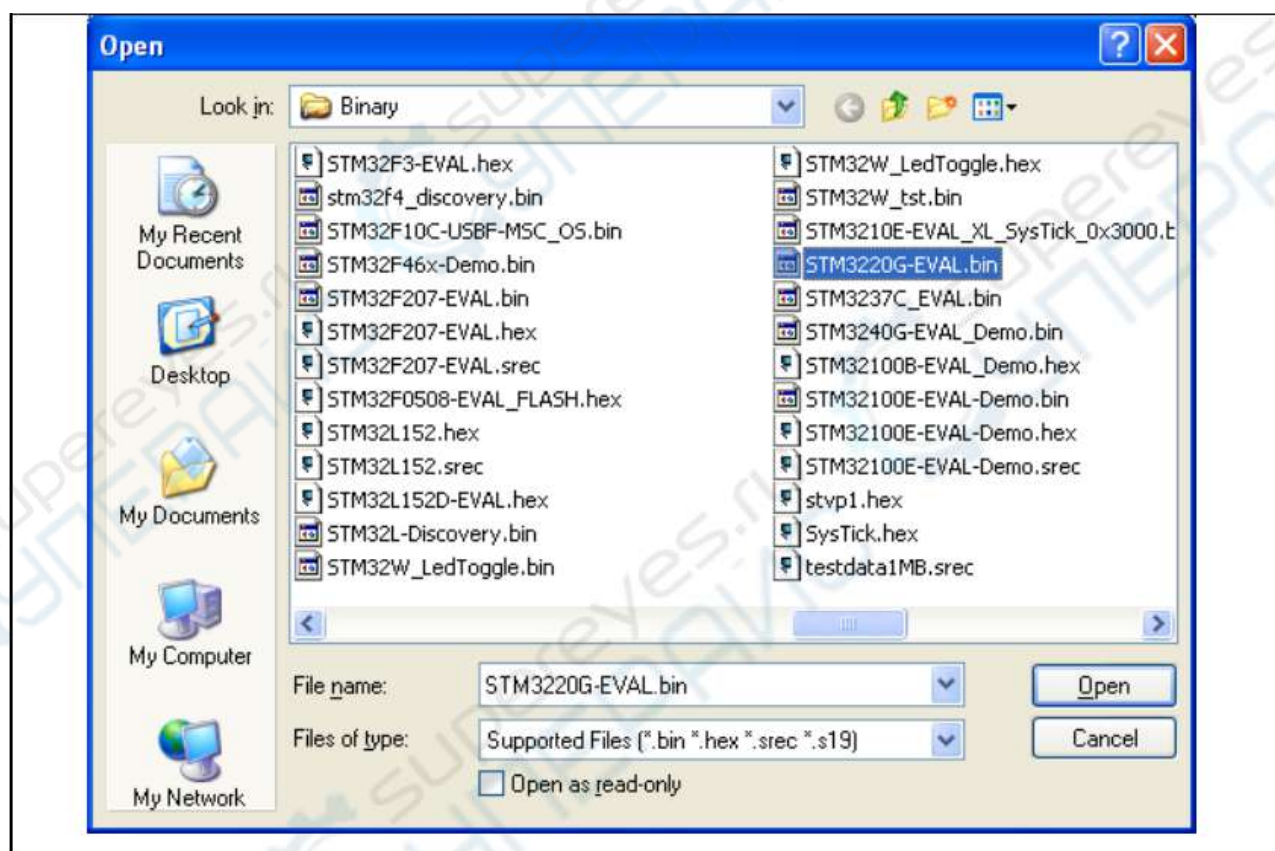


*Примечание:* Чтобы стереть сектор флэш-памяти устройств STM32 L1 со сверхнизким энергопотреблением, выберите поле памяти данных в конце списка и нажмите «Применить».

#### 4.4. Программирование устройств

Утилита STM32 ST-LINK может загружать двоичные, шестнадцатеричные или srec-файлы во флэш-память или оперативную память RAM. Чтобы сделать это, выполните следующие действия:

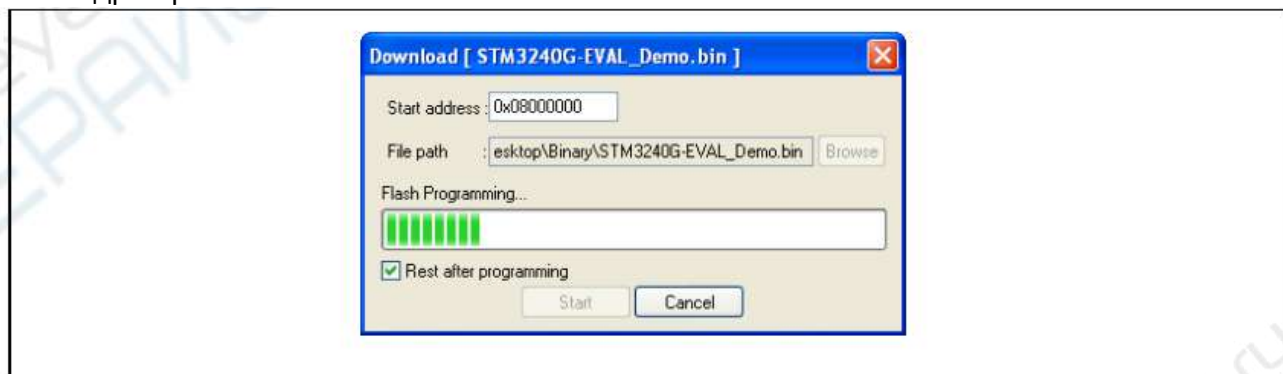
1. Нажмите на Target | Program... (или Target | Program & Verify..., если вы хотите проверить записанные данные), чтобы открыть окно Open file dialog, как показано на рисунке. Если двоичный файл уже открыт, перейдите к шагу 3.



2. Выберите двоичный файл, Intel Hex или Motorola S-record и нажмите на кнопку Открыть.



3. Укажите адрес, с которого следует начать программирование, как показано на рисунке, это может быть адрес флэш-памяти или ОЗУ.

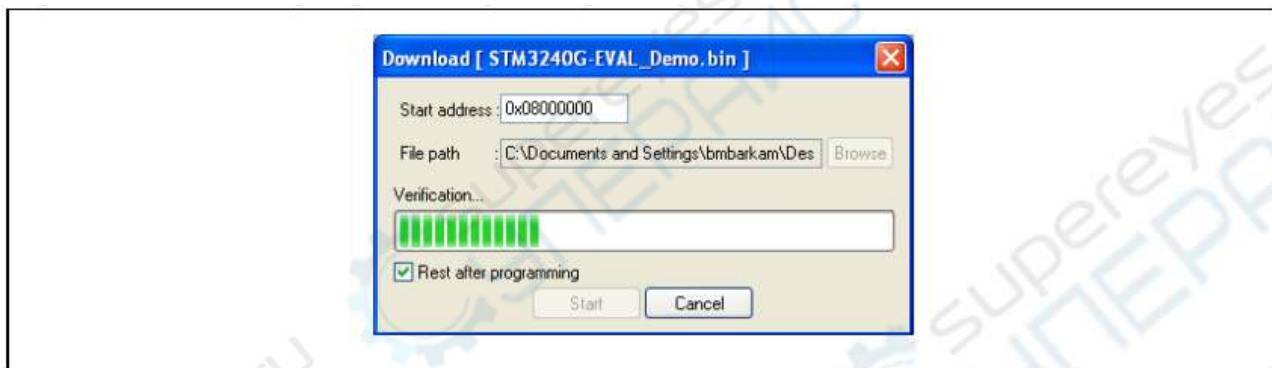


4. Наконец, нажмите на кнопку Start для начала программирования. Если на первом этапе вы выбрали Target | Program & Verify..., то по окончании программирования будет выполнена проверка (см. рис.). Если установлен флажок "Reset after programming", то будет выполнен сброс MCU.

*Примечание:* Микросхемы серий STM32 F2 и STM32 F4 поддерживают различные режимы программирования в зависимости от напряжения питания MCU. При использовании ST-LINK напряжение питания MCU должно быть указано в меню Target | Settings, чтобы можно было запрограммировать устройство в нужном режиме.

При использовании ST-LINK/V2 напряжение питания определяется автоматически.

Если устройство защищено от чтения, то защита будет отключена. Если некоторые страницы Flash-памяти защищены от записи, то при программировании защита будет отключена, а затем восстановлена.



#### 4.5. Конфигурация опционных байтов

Утилита STM32 ST-LINK Utility может настроить все опционные байты с помощью диалогового окна Option Bytes, показанном на рисунке 13, доступ к которому осуществляется по команде Target | Option Bytes.....

Диалоговое окно Option Bytes содержит следующие разделы:

**Защита от считывания:** Изменяет состояние защиты флэш-памяти от чтения.

Для устройств STM32 F0, STM32 F2, STM32 F3, STM32 F4 и STM32 L1 доступны уровни защиты от чтения:

- Уровень 0: защита от чтения отсутствует
- Уровень 1: защита памяти от чтения включена
- Уровень 2: защита памяти от чтения включена, а все отладочные функции отключены.

Для остальных устройств защита от чтения может быть только включена или выключена.

**BOR Level:** Уровень сброса напряжения. Этот список содержит порог уровня питания, который активирует/отключает сброс напряжения. Эта опция включена только при подключении к устройствам STM32 L1, STM32 F2 и STM32 F4.

Для устройств со сверхнизким энергопотреблением можно выбрать 5 программируемых порогов VBOR:

- BOR LEVEL 1: сброс порогового уровня для диапазона напряжения от 1,69 до 1,8 В.
- BOR LEVEL 2: сброс порогового уровня для диапазона напряжения от 1,94 до 2,1 В.
- BOR LEVEL 3: сброс порогового уровня для диапазона напряжения от 2,3 до 2,49 В.
- BOR LEVEL 4: сброс порогового уровня для диапазона напряжения от 2,54 до 2,74 В.
- BOR LEVEL 5: Сброс порогового уровня для диапазона напряжения от 2,77 до 3,0 В.

Для устройств STM32 F2 и STM32 F4 можно выбрать 4 программируемых порога VBOR:

- BOR LEVEL 3: Напряжение питания находится в диапазоне от 2,70 до 3,60 В.
- BOR LEVEL 2: Напряжение питания находится в диапазоне от 2,40 до 2,70 В.
- BOR LEVEL 1: Напряжение питания находится в диапазоне от 2,10 до 2,40 В.
- BOR выключен: напряжение питания находится в диапазоне от 1,62 до 2,10 В.

Конфигурации пользователя опционных байтов:

- WDG\_SW: Если установлен флажок, сторожевой таймер включен программным обеспечением. В противном случае он автоматически включается при включении питания.

- nRST\_STOP: Если флажок не установлен, при переходе в режим ожидания генерируется сброс (домен выключен напряжением 1,8 В). Если этот флажок установлен, сброс не производится при переходе в режим ожидания.

- nRST\_STDBY: если флажок не установлен, при входе в режим остановки (Stop) генерируется сброс (все часы останавливаются). Если установлен флажок, сброс не генерируется при входе в режим остановки.

- BFB2: Если флажок не установлен и если загрузочные контакты настроены на загрузку устройства с пользовательской флэш-памяти при запуске, устройство загружается из банка флэш-памяти 2; в противном случае оно загружается из банка флэш-памяти 1. Эта опция включена только при подключении к устройству, содержащему два банка флэш-памяти.

- nBoot1: Вместе с выводом BOOT0 выбирает режим загрузки:

nBOOT1 установлен/отменен и BOOT0=0 => загрузка из основной Flash-памяти.

nBOOT1 установлен и BOOT0 =1 => Загрузка из системной памяти.

nBOOT1 не отмечен и BOOT0 =1 => загрузка из встроенной SRAM".

nBoot1 доступен только для STM32 F0 и STM32 F3.

- VDDA\_Monitor: выбирает аналоговый мониторинг источника питания VDDA:

Если этот флажок установлен, супервизор источника питания VDDA включен, в противном случае он отключен.

VDDA доступен только на STM32 F0 и STM32 F3.

- nSRAM\_Parity: этот бит позволяет пользователю включить аппаратную проверку четности SRAM. Если этот флажок установлен, проверка четности SRAM отключена; в противном случае она включена.

nSRAM\_Parity доступен только в STM32 F0 и STM32 F3.

- SDADC12\_VDD\_Monitor:

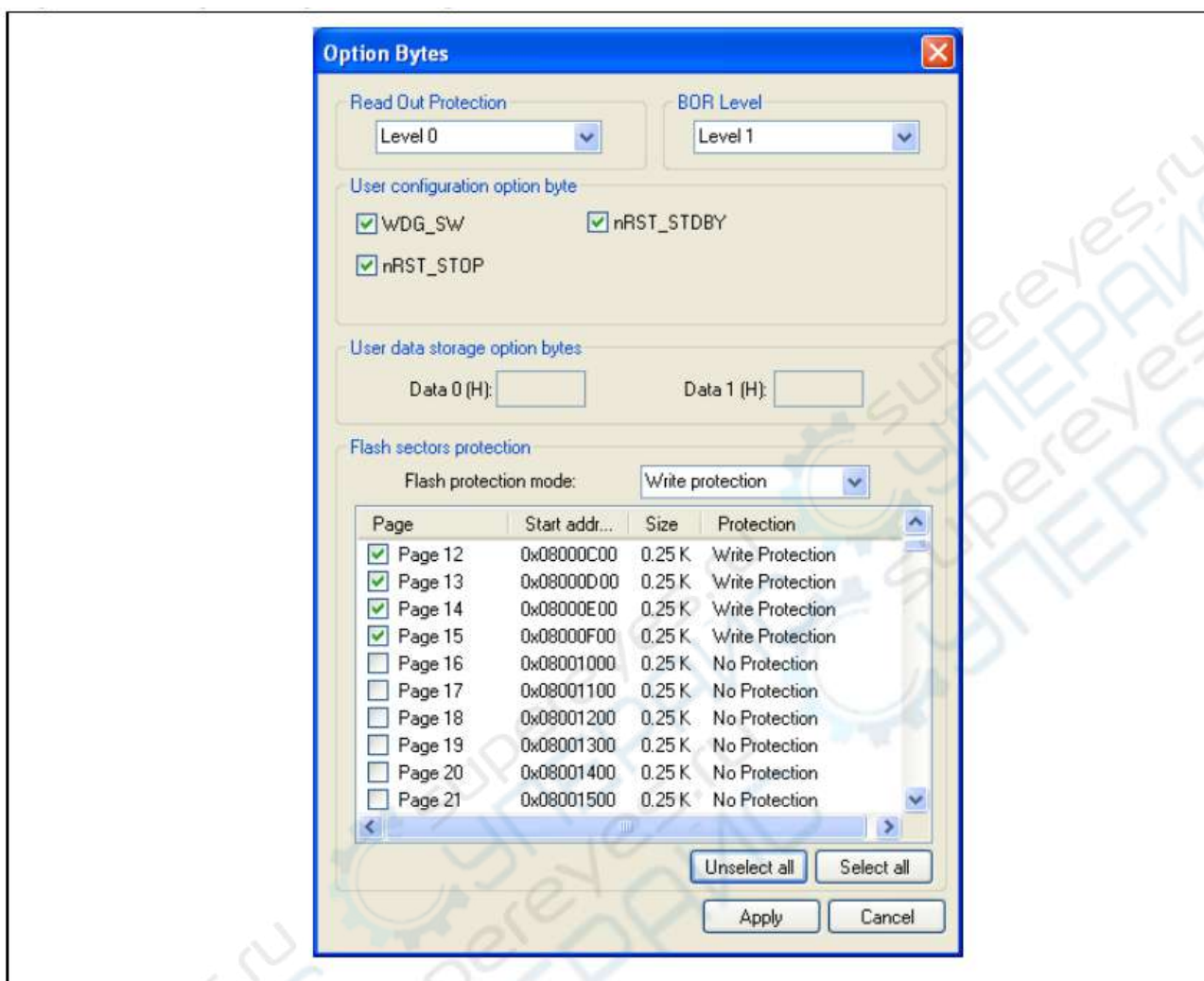
Если этот флажок установлен, супервизор источника питания SDADC12\_VDD включен, в противном случае он отключен.

SDADC12\_VDD\_Monitor доступен только на STM32F37x.

- Параметр хранения пользовательских данных в байтах (User data storage option bytes): содержит два байта для хранения пользовательских данных. Эти два байта параметров недоступны в устройствах STM32F0, STM32 F2, STM32 F3, STM32 F4 и STM32 L1.



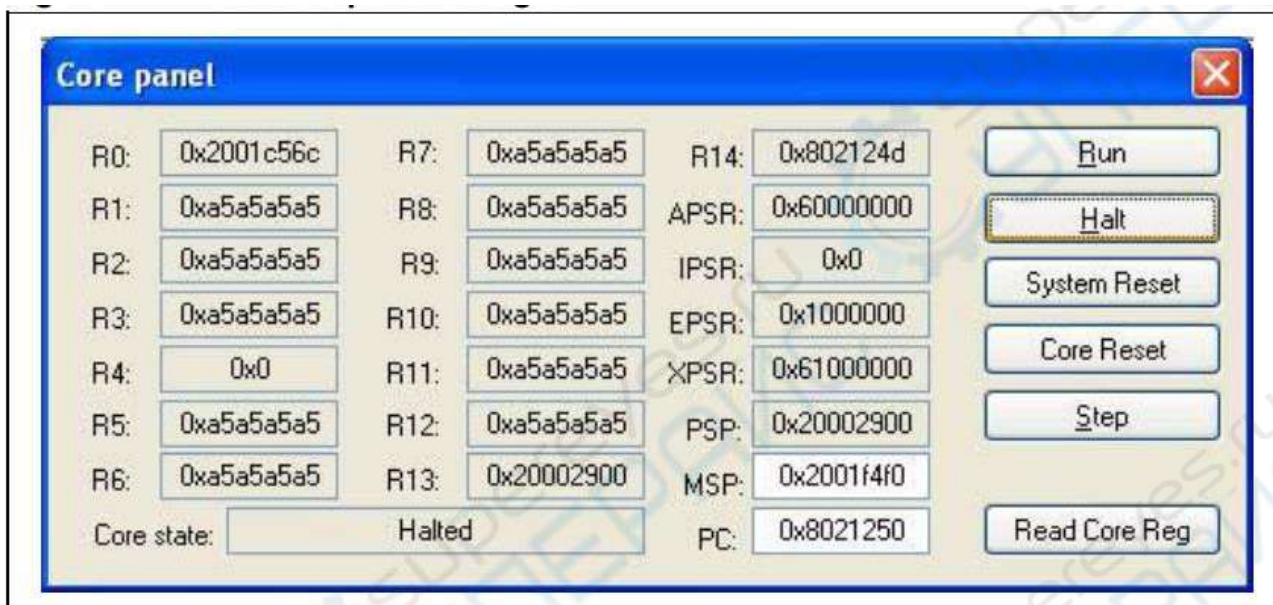
- Защита флэш-секторов (Flash Sectors Protection): В зависимости от подключенного устройства флэш-секторы группируются по определенному количеству страниц. Здесь вы можете изменить защиту от записи для каждого сектора флэш-памяти.
- Для устройств, поддерживающих функцию PCROP, можно включить/отключить защиту от чтения каждого сектора. “Режим защиты флэш-памяти” позволяет вам выбирать между защитой от чтения или записи



#### 4.6. Основные функции микроконтроллера (MCU)

В диалоговом окне Core panel, показанном на рис., отображаются значения регистров ядра Cortex™-M3. Оно также позволяет выполнять следующие действия с MCU, используя кнопки справа:

- Run: Запустить ядро.
- Halt: Остановить работу ядра.
- System Reset (Сброс системы): Отправить запрос на сброс системы.
- Сброс ядра: Сброс ядра.
- Step (шаг): Выполнение только одной пошаговой инструкции ядра.
- Read Core Reg: обновление значений регистров ядра.



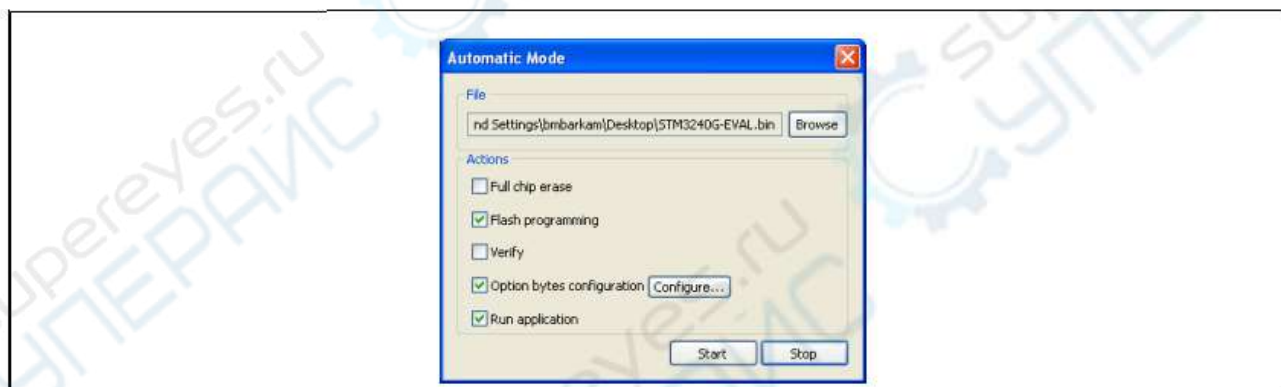
Примечание: Регистры PC и MSP могут быть изменены с этой панели.

#### 4.7. Функции автоматического режима

Диалоговое окно Automatic Mode (Автоматический режим), показанное на рис., позволяет программировать и конфигурировать устройства STM32 в цикле. Оно позволяет выполнить следующие действия над устройством STM32:

- Полное стирание микросхемы
- Программирование флэш-памяти
- Верификация
- Конфигурирование опционных байтов
- Запуск приложения

Нажатие на кнопку Start приведет к выполнению выбранных действий на подключенном устройстве STM32 и к ожиданию повторения тех же действий после отключения текущего и подключения нового устройства.



Примечание: Если пользователь отменяет выбор действия программирования флэш-памяти, когда флэш-память STM32 защищена от считывания, она автоматически снимается с защиты.

Если пользователь отменяет выбор действия программирования флэш-памяти, в то время как некоторые или все флэш-памяти STM32 защищены от записи, они будут автоматически сняты с защиты и восстановлены после операции программирования.

Для выбора опции конфигурирования байтов с помощью кнопки configure необходимо установить соединение с устройством.

Подключаемые устройства должны быть производными одного семейства STM32 и подключены в одном режиме (JTAG или SWD).

## 5. Интерфейс командной строки (CLI) STM32 ST-LINK Utility

### 5.1. Использование командной строки

В следующих разделах описывается работа с утилитой STM32 ST-LINK Utility из командной строки.

Интерфейс командной строки ST-LINK Utility находится по следующему адресу:

[Install\_Directory]\STM32 ST-LINK Utility\ST-LINK Utility\ST-LINK\_CLI.exe

#### 5.1.1 Команды подключения и работы с памятью

**-c**

**Описание:** Выбор протокола связи JTAG или SWD. По умолчанию используется протокол JTAG.

**Синтаксис:** -c [JTAG/SWD] [UR]

[UR]: Подключение к цели при сбросе.

Пример: -c SWD UR

*Примечание:* [UR] доступно только для ST-LINK/V2 и в режиме SWD.

Для режима JTAG подключение под сброс доступно начиная с версии прошивки ST-LINK V2J15Sx.

Контакт RESET разъема JTAG (вывод 15) должен быть подключен к контакту сброса устройства.

**-Q**

**Описание:** Включить тихий режим. Индикатор выполнения не отображается.

**-w8**

**Описание:** Записывает 8-битные данные по указанному адресу памяти.

**Синтаксис:** -w8 <Address> <data>.

Пример: -w8 0x20000000 0xAA

*Примечание:* -w8 поддерживает запись во Flash-память, OTP, SRAM и регистры R/W.

**-w32**

**Описание:** Записывает 32-битные данные по указанному адресу памяти.

**Синтаксис:** -w32 <Address> <data>.

Пример: -w32 0x08000000 0xAABBCCDD

*Примечание:* -w32 поддерживает запись во Flash-память, OTP, SRAM и регистры R/W.

**-r8**

**Описание:** Считывает <NumBytes> памяти.

**Синтаксис:** -r8 <Address> <NumBytes>.

Пример: -r8 0x20000000 0x100

#### 5.1.2 Основные команды

**-Rst**

**Описание:** Сброс системы.

**Синтаксис:** -Rst

**-HardRst**

**Описание:** Аппаратный сброс.

**Синтаксис:** -HardRst

*Примечание:* команда -HardRst доступна только для ST-LINK/V2. Контакт RESET разъема JTAG (вывод 15) должен быть подключен к контакту сброса устройства.

#### **-Run**

**Описание:** Устанавливает счетчик программы и указатель стека, заданные в пользовательском приложении, и выполняет операцию запуска. Это удобно, если пользовательское приложение загружено со смещением (например, 0x08003000). Если адрес не указан, то используется 0x08000000.

**Синтаксис:** -Run [<Address>].

**Пример:** -run 0x08003000

#### **-Halt**

**Описание:** Останавливает работу ядра.

**Синтаксис:** -Halt

#### **-Step**

**Описание:** Выполняет инструкцию Step ядра.

**Синтаксис:** -Step

#### **-SetBP**

**Описание:** Устанавливает программную или аппаратную точку останова по определенному адресу. Если адрес не указан, используется значение 0x08000000.

**Синтаксис:** -SetBP [<Address>].

**Пример:** -SetBP 0x08003000

#### **-ClrBP**

**Описание:** Очищает все аппаратные точки останова, если таковые имеются.

**Синтаксис:** -ClrBP

#### **-CoreReg**

**Описание:** Считывает регистры ядра.

**Синтаксис:** -CoreReg

#### **-SCore**

**Описание:** Определяет состояние ядра.

**Синтаксис:** -SCore

### **5.1.3 Флэш-команды**

#### **-ME**

**Описание:** Выполняет операцию полной очистки микросхемы.

**Синтаксис:** -ME

#### **-SE**

**Описание:** Очистка сектора (секторов) флэш-памяти.

**Синтаксис:** -SE <Start\_Secto> [<End\_Sector>]

**Пример:**

-SE 0 => Erase sector 0

-SE 2 12 => Erase sectors from 2 to 12

\*Для серии STM32L следующая команда cmd стирает данные eeprom:

-SE ed1 => Erases data eeprom at 0x08080000

-SE ed2 => Erases data eeprom at 0x08081800

#### **-P**



**Описание:** Загрузите двоичный файл, Intel Hex или Motorola S-record в память устройства без проверки. Для форматов hex и srec адрес имеет значение.

**Синтаксис:** -P <File\_Path> [<Address>]

Примеры:

-P "C:\file.srec"  
-P "C:\file.bin" 0x08002000  
-P "C:\file.hex"

**Примечание:** Серии STM32 F2 и STM32 F4 поддерживают различные режимы программирования в зависимости от напряжения питания MCU. При использовании ST-LINK/V2 напряжение питания определяется автоматически. Таким образом, выбирается правильный режим программирования. При использовании ST-LINK по умолчанию выбирается 32-разрядный режим программирования.

Если устройство защищено от чтения, то защита будет отключена. Если некоторые страницы Flash-памяти защищены от записи, то при программировании защита будет отключена, а затем восстановлена.

**-V**

**Описание:** Проверяет успешность выполнения операции программирования.

**Синтаксис:** -V

Пример: -P "C:\file.srec" -V

#### 5.1.4 Команды опционных байтов

**-rOB**

**Описание:** Отображает все опционные байты.

**Синтаксис:** -rOB

**-OB**

**Описание:** Конфигурирует опционные байты. Эта команда:

- устанавливает уровень защиты от считывания на уровень 0 (защита отсутствует)
- устанавливает опцию IWDG\_SW в '1' (сторожевой таймер включается программно)
- устанавливает опцию nRST\_STOP в '0' (сброс генерируется при переходе в режим ожидания)
- устанавливает опционный байт Data0
- устанавливает байт опционный байт Data1.

**Синтаксис:**

-OB [RDP=<Level>][BOR\_LEV=<Level>][IWDG\_SW=<Value>]  
[nRST\_STOP=<Value>][nRST\_STDBY=<Value>][BFB2=<Value>]  
[nBoot1=<Value>][nSRAM\_Parity=<Value>][Data0=<Value>]  
[Data1=<Value>][WRP=<Value>][WRP2=<Value>]  
[WRP3=<Value>]

Пример:

-OB RDP=0 IWDG\_SW=1 nRST\_STOP=0  
Data0=0xAA Data1=0xBC

#### Описание параметров команды опционного байта

RDP=<Level>:

RDP=<Level> устанавливает уровень защиты флэш-памяти от чтения.

Значение <Level> может быть одним из следующих уровней:

0: Защита отключена

1: Защита включена



2: Защита включена (функции отладки и загрузки в SRAM ОТКЛЮЧЕНЫ)

Примечание: Уровень 2 доступен только для серий STM32 F0, STM32 F2, STM32 F3, STM32 F4 и STM32 L1

**BOR\_LEV=<Level>:**

BOR\_LEV устанавливает пороговый уровень сброса напряжения.

*Для серии STM32 L1:*

0: BOR ВЫКЛ, диапазон напряжения от 1,45 до 1,55 В.

1: диапазон напряжения от 1,69 до 1,8 В.

2: диапазон напряжения от 1,94 до 2,1 В.

3: диапазон напряжения от 2,3 до 2,49 В.

4: диапазон напряжения от 2,54 до 2,74 В.

5: диапазон напряжения от 2,77 до 3,0 В.

*Для серий STM32 F2 и STM32 F4:*

0: BOR ВЫКЛ, диапазон напряжения от 1,8 до 2,10 В.

1: диапазон напряжения от 2,10 до 2,40 В.

2: диапазон напряжения от 2,40 до 2,70 В.

3: диапазон напряжения от 2,70 до 3,60 В.

**IWDG\_SW=<Value>:**

Значение <Value> должно быть равно 0 или 1:

0: Аппаратно-независимый сторожевой таймер

1: Программно-независимый сторожевой таймер

**nRST\_STOP=<Value>:**

Значение <Value> должно быть 0 или 1:

0: Сброс генерируется, когда процессор переходит в режим останова.

1: Сброс не генерируется.

**nRST\_STDBY=<Value>:**

Значение <Value> должно быть равно 0 или 1:

0: Сброс генерируется, когда процессор переходит в режим ожидания.

1: Сброс не генерируется.

**BFB2=<Value>:**

Значение <Value> должно быть 0 или 1:

0: Загрузка из банка Flash 2, если загрузочные контакты установлены в положение загрузки из пользовательской Flash. (по умолчанию)

1: Загрузка из банка флэш-памяти 1, когда загрузочные контакты установлены в положение загрузки из пользовательской флэш-памяти (по умолчанию).

*Примечание:* BFB2 доступен только в устройствах, содержащих два банка Flash.

**nBoot1=<Value>:**

Значение <Value> должно быть 0 или 1:

Вместе с выводом BOOT0 выбирает режим загрузки:

nBOOT1 = 0 or 1 and BOOT0 = 0 => Загрузка из основной Flash-памяти.

nBOOT1 = 1 and BOOT0 = 1 => Загрузка из системной памяти.

nBOOT1 = 0 and BOOT0 = 1 => Загрузка из встроенной SRAM.

*Примечание:* nBoot1 доступен только для STM32 F0 и STM32 F3.

**VDDA=<Value>:**

Выбор аналогового мониторинга на источнике питания VDDA.

Значение <Value> должно быть равно 0 или 1.

*Примечание:* VDDA доступен только для STM32 F0 и STM32 F3.

**nSRAM\_Parity=<Value>:**

Этот бит позволяет пользователю включить аппаратную проверку четности SRAM.

Значение <Value> должно быть 0 или 1:

*Примечание:* функция nSRAM\_Parity доступна только для STM32 F0 и STM32 F3.

**SDADC12\_VDD=<Value>:**

Выбирает аналоговый контроль (сравнение с напряжением V<sub>garr</sub> 1,2 В) на источнике питания DADC12\_VDD.

Значение <Value> должно быть 0 или 1:

*Примечание:* SDADC12\_VDD доступен только в устройствах STM32F37x.

**Data0=<Value>:**

Data0 устанавливает опционный байт Data0.

Значение <Value> должно находиться в диапазоне [0..0xFF].

*Примечание:* Недоступно для устройств STM32 F0, STM32 F2, STM32 F3, STM32 F4 и STM32 L1.

**Data1=<Value>:**

Data1 устанавливает опционный байт Data1.

Значение <Value> должно находиться в диапазоне [0..0xFF].

*Примечание:* Недоступно для устройств STM32 F0, STM32 F2, STM32 F3, STM32 F4 и STM32 L1.

**SPRMODE=<Value>:**

Выбор режима защиты битов WPRi:

Значение <Value> должно быть 0 или 1:

0: Биты WPRi используются для защиты от записи в секторе i (по умолчанию).

1: Биты WPRi используются для PCROP-защиты (защиты от чтения) сектора i (Сектор).

*Примечание:* Доступно только для устройств, поддерживающих функцию PCROP.

**WRP=<Value>:**

WRP включает/выключает защиту от записи секторов флэш-памяти MCU.

Каждый бит включает/выключает защиту от записи одного или нескольких секторов в зависимости от подключенного устройства.

Для устройств STM32 L1 WRP[i] = 0: сектор(ы) Flash защищен(ы).

Для других устройств WRP[i] = 1: сектор(ы) флэш-памяти защищен(ы).

Этой команды достаточно для включения/выключения защиты всех секторов Flash, за исключением STM32 L1 High density, где требуются команды WRP2 и WRP3.

Для STM32 серии F4 каждый бит WRP включает/выключает защиту от записи одного сектора.

Значение <Value> должно находиться в диапазоне [0..0xFFFFFFFF].

*Примечание:* Для устройств, поддерживающих функцию PCROP, WRP управляет защитой секторов флэш-памяти MCU от чтения при SPRMODE = 1.

**WRP2=<Value>:**

WRP2 доступен только для устройств STM32 L1 средней плотности плюс и высокой плотности для включения/выключения защиты секторов Flash со страницы 512 по 1023.

Значение <Value> должно находиться в диапазоне [0..0xFFFFFFFF].

*Примечание:* Для устройств, поддерживающих функцию PCROP, WRP управляет защитой секторов флэш-памяти MCU от чтения при SPRMODE = 1.

**WRP3=<Value>:**

WRP3 доступен только для устройств STM32 L1 высокой плотности для включения/выключения защиты секторов Flash со страницы 1024 по 1535.

Значение <Value> должно находиться в диапазоне [0..0xFFFFFFFF].

*Примечание:* 1 Все перечисленные параметры должны быть в шестнадцатеричном формате

**5.1.5 Коды возврата ST-LINK\_CLI**

В случае возникновения ошибки при выполнении команд ST-LINK\_CLI код возврата (Errorlevel) будет больше 0.

В следующей таблице приведены коды возврата ST-LINK\_CLI:

Код возврата	Команды	Ошибка
1	All	Ошибка в аргументах команды.
2	All	Проблема с подключением.
3	All	Команда недоступна для подключенной цели.
4	-w8, -w32	Произошла ошибка при записи данных в указанный адрес памяти.
5	-r8, r32	Невозможно считать память с указанного адреса памяти.
6	-rst, -HardRst	Невозможно сбросить MCU.
7	-Run	Не удалось запустить приложение.
8	-halt	Не удалось остановить ядро.
9	-STEP	Не удалось выполнить один шаг инструкции.
10	-SetBP	Не удалось установить/очистить точку останова.
11	-ME, -SE	Невозможно стереть один или несколько секторов Flash.
12	-P, -V	Ошибка программирования опциональных байтов. Ошибка программирования/проверки флэш-памяти.
13	-OB	Ошибка программирования опциональных байтов.

**6. Эксплуатация**

ST-LINK / V2 является внутрисхемным отладчиком / программатором для семейства микроконтроллеров STM8 и STM32. Однопроводный интерфейсный модуль (SWIM) и JTAG / SWD интерфейсы облегчают обмен данными с любыми STM8 или STM32 микроконтроллерами, расположенными на плате устройства.

В дополнение к тем же функциям, что и ST-LINK/V2, ST-LINK/V2-ISOL обладает цифровой изоляцией между ПК и целевой платой применения. Он также выдерживает напряжения до 1000 В RMS.

Полноскоростной интерфейс USB обеспечивает связь с ПК и:

- STM8 устройством через программное обеспечение ST Visual Development (STVD) или ST Visual Program (STVP) (они доступны от STMicroelectronics).

- STM32 устройством с помощью интегрированных сред разработки Atollic, IAR , Keil и TASKING.



ST-LINK/V2



ST-LINK/V2-ISOL

## 6.1 Конфигурация

ST-LINK/V2 спроектирован на базе микроконтроллера STM32F103C8, который включает в себя высокопроизводительное ядро ARM Cortex-M3. Как показано на рисунке, ST-LINK/V2 обеспечивает два разъема:

- разъем STM32 для интерфейса JTAG / SWD и SWV
  - разъем STM8 для интерфейса SWIM
- ST-LINK/V2-ISOL обеспечивает один разъем для интерфейсов STM8 SWIM, STM32 JTAG / SWD и SWV.



A = Разъем STM32 JTAG и SWD

B = Разъем SWIM STM8

C = Разъем STM8 SWIM, STM32 JTAG и SWD

D = Индикатор активности связи



## 6.2 Подключение к приложениям STM8

Для отладки микроконтроллеров STM8 программатор/отладчик ST-LINK/V2 можно подключить к целевой плате с помощью двух различных кабелей, в зависимости от разъема, доступного на плате приложения.

Эти кабели представляют собой:

- плоская лента SWIM со стандартным разъемом ERNI на одном конце
- SWIM-кабель с двумя 4-контактными разъемами 2,54 мм или SWIM-кабель с раздельными проводами

### 6.2.1 Стандартное соединение ERNI с плоской лентой SWIM

На рисунке показано, как подключить ST-LINK/V2, если на плате приложения присутствует стандартный 4-контактный SWIM-разъем ERNI.



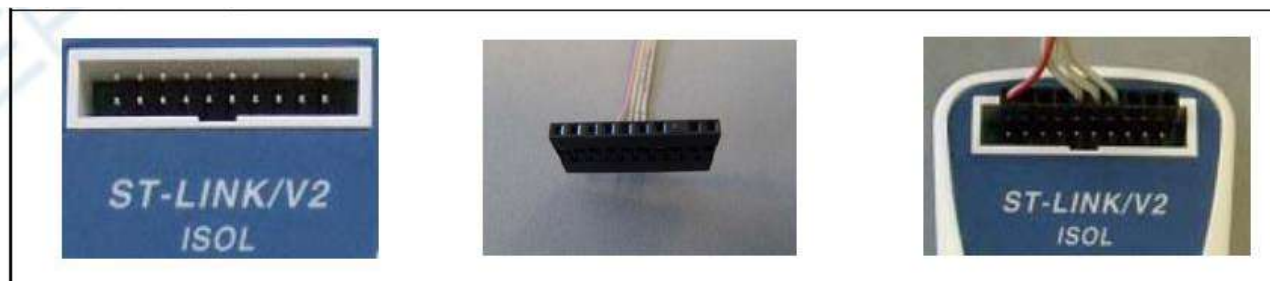
A = Целевая плата с разъемом ERNI

B = Кабель с разъемом ERNI на одном конце

C = Разъем для подключения SWIM STM8

На рисунке показано, что контакт 16 отсутствует на целевом разъеме ST-LINK/V2-ISOL. Этот недостающий контакт используется в качестве защитного ключа на разъеме кабеля, чтобы гарантировать соединение кабеля SWIM в правильном положении, используемых для обоих кабелей SWIM и JTAG.





### 6.2.2 LOW COST соединение SWIM

На рисунке показано, как подключить ST-LINK/V2, если на плате приложения имеется 4-контактный SWIM-разъем с шагом 2,54 мм.



A = Целевая плата с 4-контактным разъемом с шагом 2,54мм

B = Кабель с разъемом SWIM или кабель с отдельными проводами

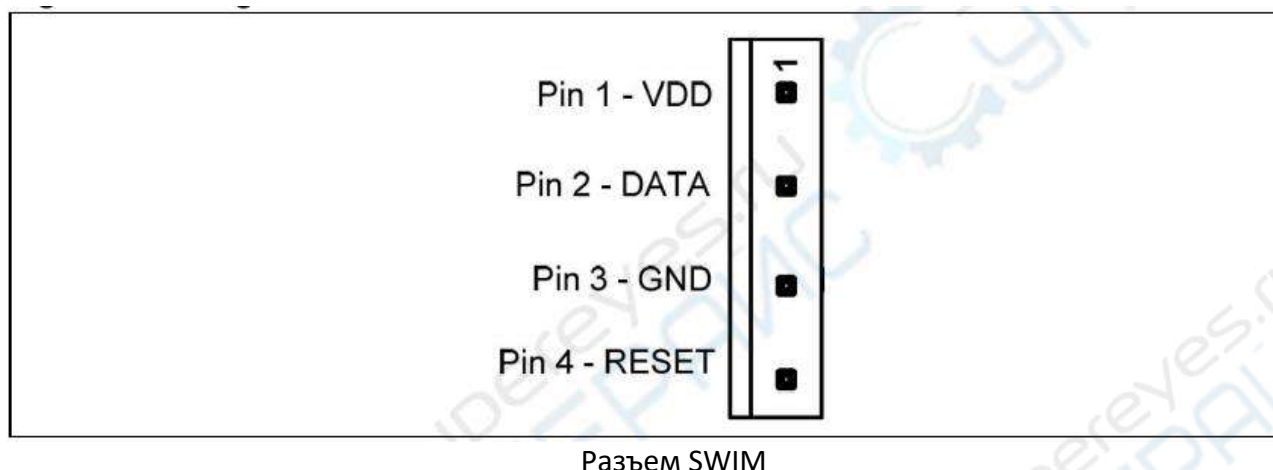
C = Разъем для подключения SWIM STM8

### 6.2.3 Сигналы SWIM соединения для ST-LINK/V2

В таблице приведены названия сигналов, функции и сигналы целевого соединения с помощью SWIM кабеля для ST-LINK/V2.

№ вывода	Наименование	Функция	Целевое соединение
1	VDD	Target VCC(1)	MCU VCC
2	DATA	SWIM	MCU SWIM вывод
3	GND	GROUND	GND
4	RESET	RESET	MCU RESET вывод

1. Питание от платы приложения подключено к панели отладки и программирования STLINK/V2, чтобы обеспечить совместимость сигналов между обеими платами.



В таблице приведены названия сигналов, функции и сигналы целевого соединения с помощью SWIM кабеля для ST-LINK/V2-ISOL. Поскольку отдельный кабельный провод SWIM имеет независимые разъемы для всех контактов на одной стороне, можно подключить ST-LINK/V2-ISOL к прикладной плате без стандартного разъема SWIM. На этом кабеле все сигналы ссылаются на определенный цвет, чтобы облегчить подключение к целевому устройству.

Цвет	Наименование контакта кабеля	Функция	Целевое соединение
Красный	TVCC	Target VCC(1)	MCU VCC
Зеленый	UART-RX	Неиспользуемые	Зарезервировано (2) (не подключен к целевой плате)
Синий	UART-TX		
Желтый	BOOT0		
Оранжевый	SWIM	SWIM	MCU SWIM вывод
Черный	GND	GROUND	GND
Белый	SWIM-RST	RESET	MCU RESET вывод

1. Питание от платы приложений подключено к панели отладки и программирования STLINK/V2, чтобы обеспечить совместимость сигналов между обеими платами.

2. BOOT0, UART-TX и UART-RX зарезервированы для будущих разработок. TVCC, SWIM, GND и SWIM-RST могут быть подключены к разъему с шагом 2,54 мм или штыревому разъему, доступному на целевой плате.

### 6.3 Подключение к приложениям STM32

Для отладки микроконтроллеров STM32 программатор/отладчик ST-LINK/V2 необходимо подключить к приложению, используя стандартный 20-контактный плоский кабель JTAG.

В таблице приведены названия сигналов, функции и сигналы целевого соединения JTAG.

№ контакта	ST-LINK/V2 разъем (CN3)	ST-LINK/V2 функция <sup>2</sup>	Целевое соединение (JTAG)	Целевое соединение (SWD)
1	VAPP	Target VCC	MCU VDD <sup>(1)</sup>	MCU VDD <sup>(1)</sup>
2				
3	TRST	JTAG TRST	JNTRST	GND <sup>(2)</sup>
4	GND	GND	GND <sup>(3)</sup>	GND <sup>(3)</sup>
5	TDI	JTAG TDO	JTDI	GND <sup>(2)</sup>
6	GND	GND	GND <sup>(3)</sup>	GND <sup>(3)</sup>
7	TMS_SWDIO	JTAG TMS, SW IO	JTMS	SWDIO
8	GND	GND	GND <sup>(3)</sup>	GND <sup>(3)</sup>
9	TCK_SWCLK	JTAG TCK, SW CLK	JTCK	SWCLK
10	GND	GND	GND <sup>(3)</sup>	GND <sup>(3)</sup>
11	NC	Не подключен	Не подключен	Не подключен
12	GND	GND	GND <sup>(3)</sup>	GND <sup>(3)</sup>
13	TDO_SWO	JTAG TDI, SWO	JTDO	TRACESWO <sup>(4)</sup>
14	GND	GND	GND <sup>(3)</sup>	GND <sup>(3)</sup>
15	NRST	NRST	NRST	NRST
16	GND	GND	GND <sup>(3)</sup>	GND <sup>(3)</sup>
17	NC	Не подключен	Не подключен	Не подключен
18	GND	GND	GND <sup>(3)</sup>	GND <sup>(3)</sup>
19	VDD	VDD (3.3V) <sup>(5)</sup>	Не подключен	Не подключен
20	GND	GND	GND <sup>(3)</sup>	GND <sup>(3)</sup>

1. Питание от платы приложений подключено к панели отладки и программирования STLINK/V2, чтобы обеспечить совместимость сигналов между обеими платами.

2. Подключите к GND для уменьшения шума на кабеле.

3. По крайней мере, один из этих контактов должен быть подключен к земле для правильного поведения (рекомендуется использовать все из них).

4. Необязательно: для отслеживания последовательного прохода (SWV)

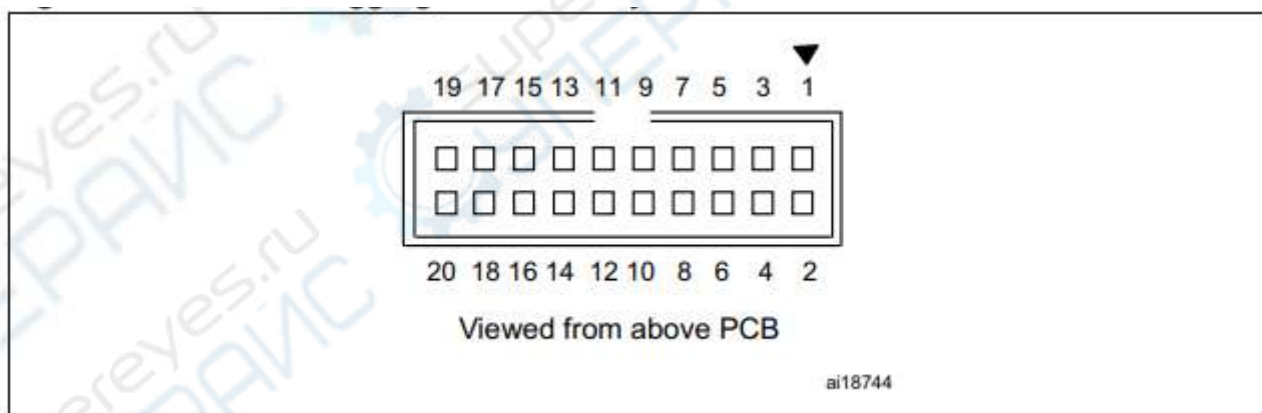
5. Доступно только для ST-LINK / V2 и не подключено к ST-LINK / V2 / OPTO.

На рисунке показано, как подключить ST-LINK/V2 к целевому устройству с помощью кабеля JTAG.



A = Целевая плата с разъемом JTAG  
 B = JTAG / SWD 20-жильный плоский кабель  
 C = STM32 Разъем JTAG и SWD

Ссылка на разъем, необходимый на целевой прикладной плате: 2x10C header wrapping 2x40C H3/9.5 (pitch 2.54) - HED20 SCOTT PHSD80.



*Примечание:* Для low cots приложений или в тех случаях, когда стандартный 20-контактный разъем с шагом 2,54 мм занимает слишком большую площадь, можно использовать решение Tag-Connect для экономии средств и места на прикладной плате. Адаптер Tag-Connect и кабель обеспечивают простое и надежное подключение ST-LINK/V2 или ST-LINK/V2-ISOL к печатной плате, не требуя наличия ответного компонента на прикладной плате.

#### 6.4 Светодиодные индикаторы состояния ST-LINK/V2

Индикатор с надписью "COM" в верхней части ST-LINK/V2 показывает состояние ST-LINK/V2 (независимо от типа подключения).

Когда:

- Светодиод мигает КРАСНЫМ: выполняется первое подключение USB к ПК.



- Индикатор горит КРАСНЫМ: связь между ПК и ST-LINK/V2 установлена (завершение перечисления).
- Светодиод мигает зеленым/КРАСНЫМ: происходит обмен данными между объектом и ПК.
- Индикатор горит ЗЕЛЕНЫМ: последнее подключение прошло успешно.
- Индикатор горит ОРАНЖЕВЫМ цветом: сбой связи ST-LINK/V2 с целевым объектом.

## 7. Конфигурация программного обеспечения

### 7.1 Обновление прошивки ST-LINK/V2

ST-LINK/V2 имеет встроенный механизм обновления прошивки. Поскольку прошивка развивается в течение всей жизни продукта ST-LINK/V2 (добавляя новую функциональность, исправляются ошибки, дополняется поддержка новых семейств микроконтроллеров ...), рекомендуется периодически посещать сайт [st.com/stlinkv2](http://st.com/stlinkv2), чтобы скачать новейшую версию прошивки.

### 7.2 Разработка приложений STM8

Обратитесь к ST toolset Pack24 Patch 1 или более поздним версиям, который включает ST Visual Develop (STVD) и ST Visual Programmer (STVP).

### 7.3 Разработка приложений STM32 и программирование Flash

Сторонние компиляторы, Atollic TrueSTUDIO, IAR EWARM, Keil MDK-ARM и TASKING VX-toolset поддерживают работу с ST-LINK/V2 в соответствии с версиями, приведенными в таблице или в самой последней доступной версии

**Сторонние инструментальные средства поддерживающие ST-LINK / V2**

Разработчики IDE	Инструмент	Версия
Atollic	TrueSTUDIO	2.1
IAR	EWARM	6.20
Keil	MDK-ARM	4.20
TASKING	VX-toolset for ARM Cortex-M	4.0.1

Для работы ST-LINK/V2 требуется специальный USB драйвер. Если набор инструментов установлен автоматически, файл `stlink_winusb.inf` устанавливается в `<WINDIR> / inf` (где `<WINDIR>` обычно `C: / Windows`).

Если установка набора инструментов не была установлена автоматически, драйвер можно найти на [st.com](http://st.com):

1. Зайдите на сайт [st.com](http://st.com).
2. В поиска введите и найдите ST-LINK/V2.
3. Перейдите на вкладку Tools & Software и скачайте нужный драйвер.

Дополнительную информацию о сторонних инструментах можно найти на следующих веб-сайтах:

- [atollic.com](http://atollic.com)
- [iar.com](http://iar.com)
- [keil.com](http://keil.com)
- [tasking.com](http://tasking.com)