



toweroff's SIMM Flash Progger v3

Руководство пользователя v3.2

©toweroff, 2011

Введение

Программатор предназначен для чтения, стирания, программирования микросхем памяти 32-разрядных модулей стандарта SIMM80;

Чтения, стирания, программирования микросхем AT89S52, AT89S53, AT89S8252, AT89S8253 и других через интерфейс SPI внутрисхемно (ISP - In System Programming);

Чтения и программирования микросхем памяти через интерфейс I2C

Технические характеристики

1. Габариты 172x77x25мм
2. Интерфейс связи USB
3. Не имеет внешнего источника питания
4. Автоматическое определение типа модуля и алгоритма записи в зависимости от используемых в модуле микросхем
5. Сверка содержимого модулей с базой конфигурационного файла по трем хэсам - MD5, SHA1, AGI CRC, возможно самостоятельное добавление/изменение/удаление хэшей
6. ISP программирование - от 4 до 20сек, в зависимости от типа микросхемы
7. Трехцветная индикация работы программатора с возможностью изменения для каждого режима работы

Комплект поставки

1. Программатор
2. Переходник для интерфейса I2C
3. Кабель USB
4. Программное обеспечение

Программное обеспечение

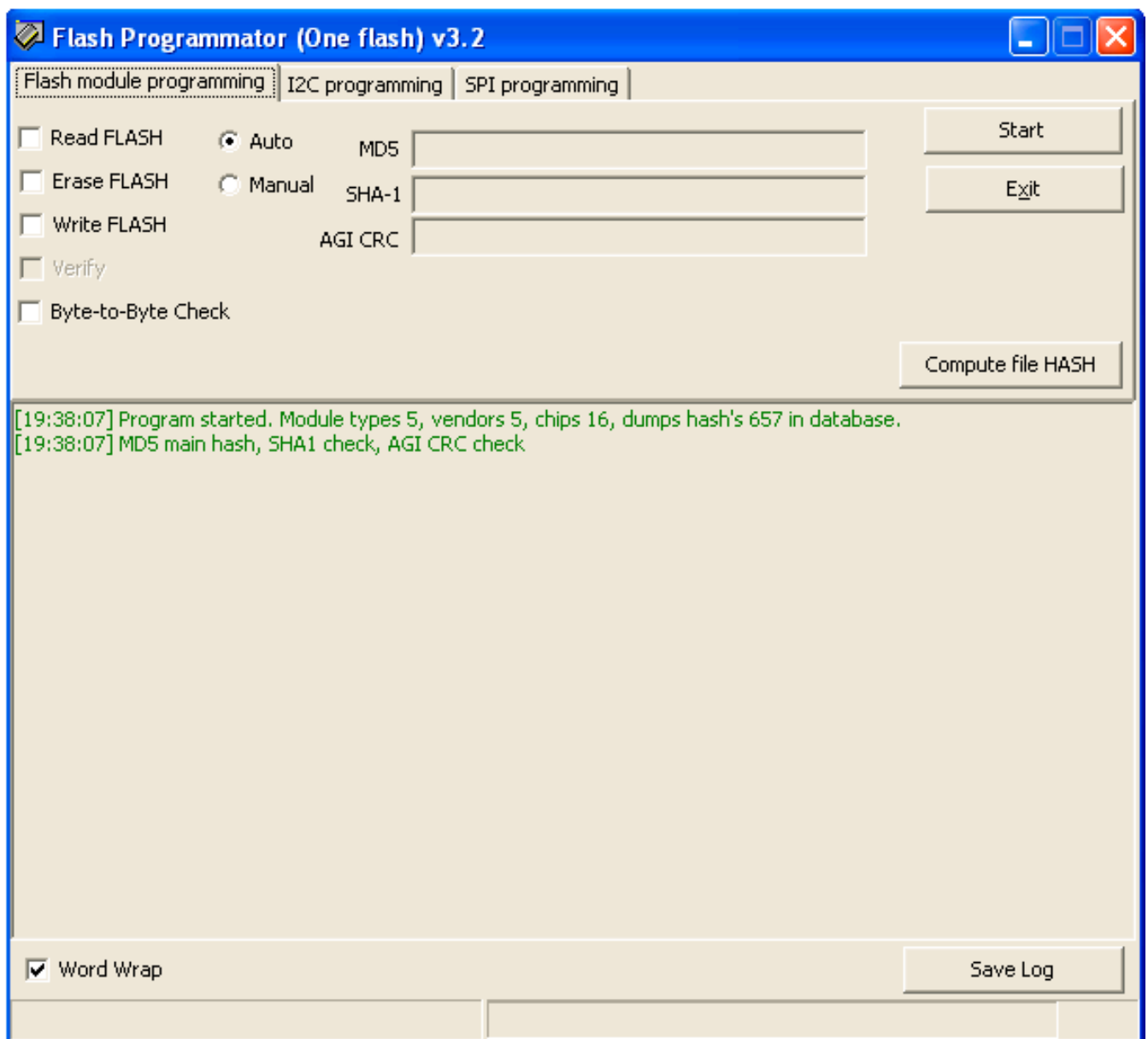
1. Установка

Запустите файл **setup_FP1_v3.2.exe**. Выберите папку установки (в дальнейшем - [INSTALL_DIR])

Подключите программатор. Система обнаружит новое устройство. Укажите папку драйвера - [INSTALL_DIR]\Driver

2. Работа с программатором

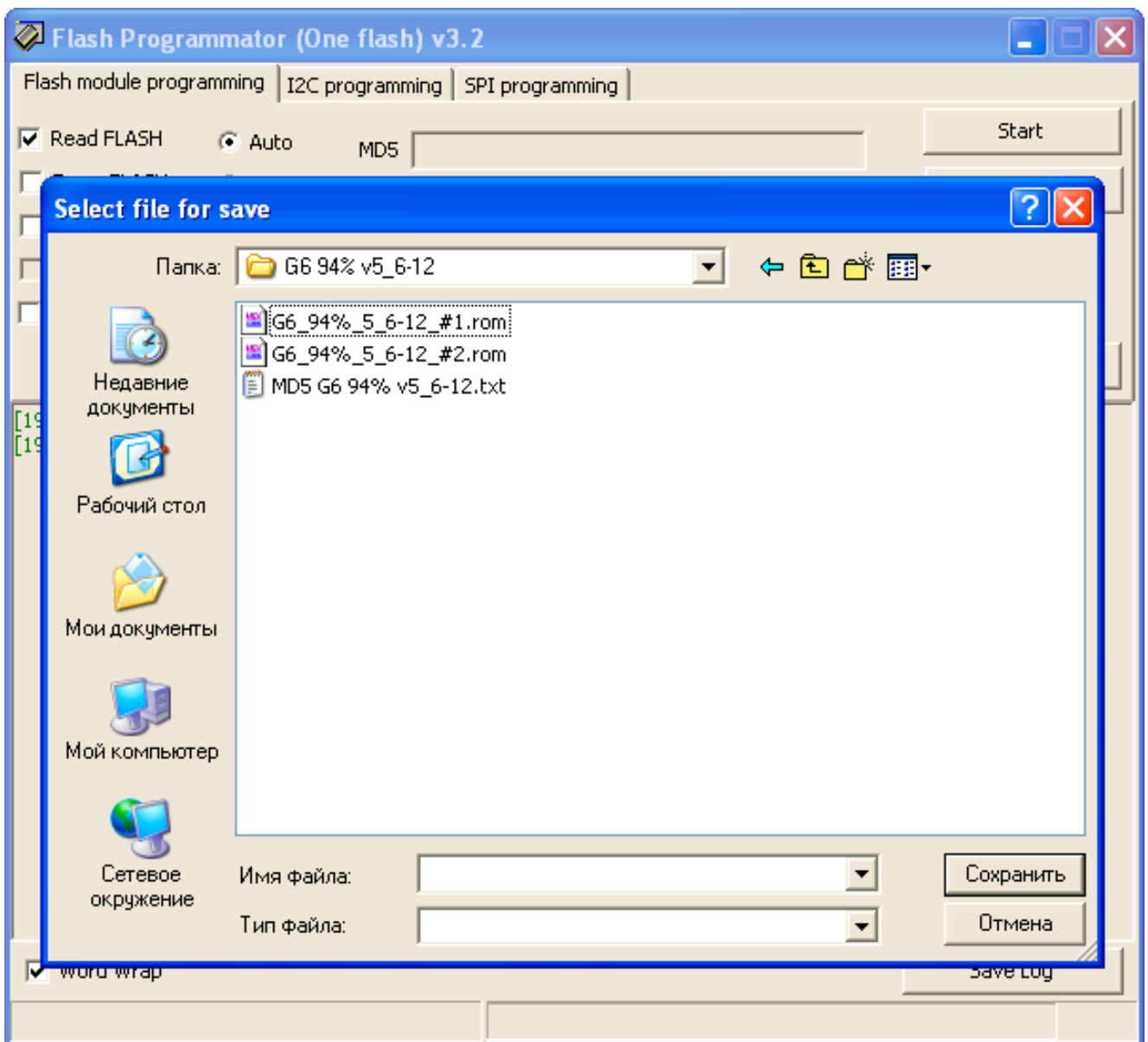
После запуска программы откроется примерно такое окно (все цифры примерные и зависят от конфигурации INI файла)



Вкладка Flash Module Programming

Read Flash

Выбор данного пункта открывает диалоговое окно, в котором необходимо указать, куда будет сохранено содержимое флеш-модуля:

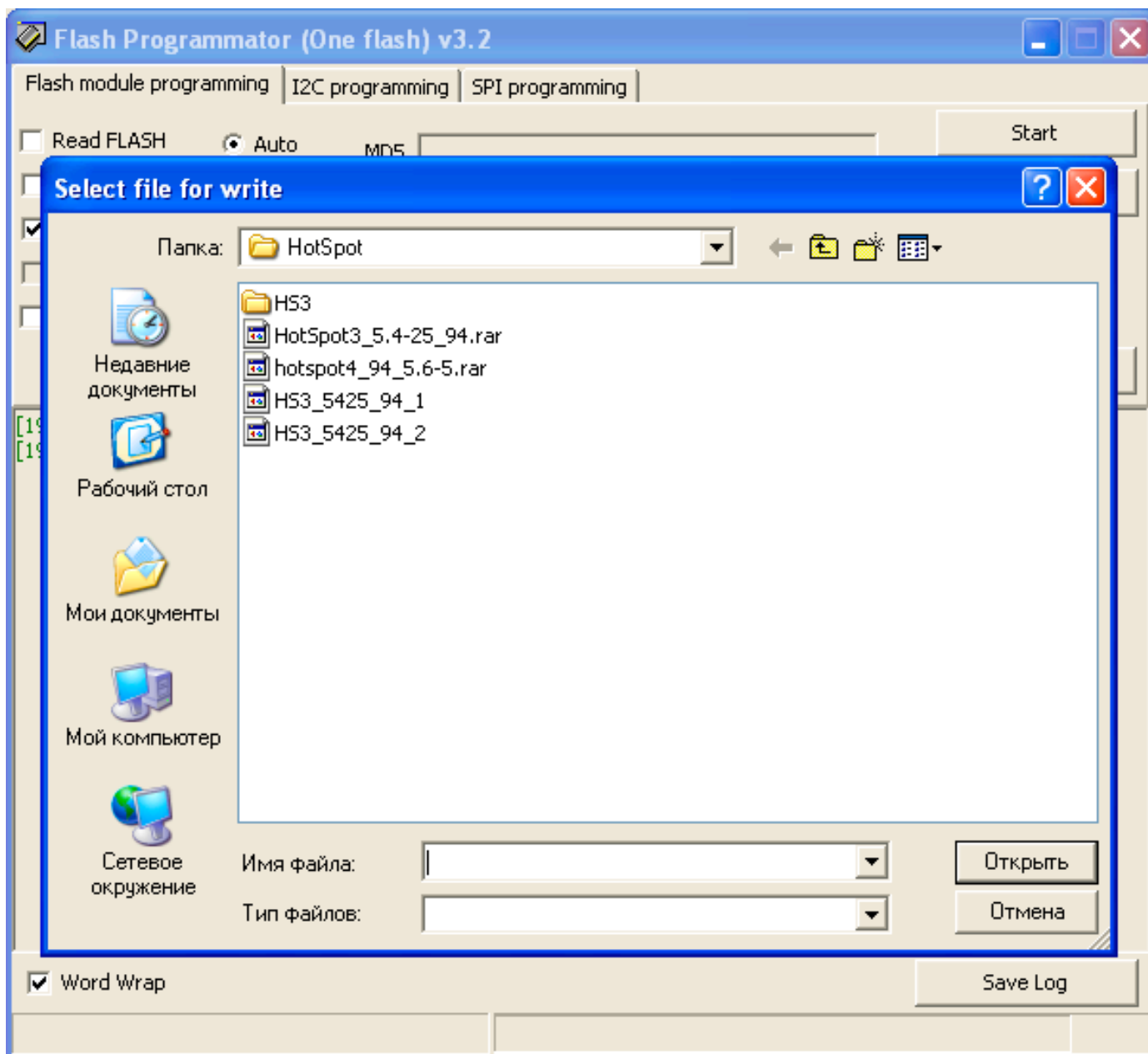


Erase Flash

Выбор данного пункта включает в список заданий стирание микросхем SIMM модуля

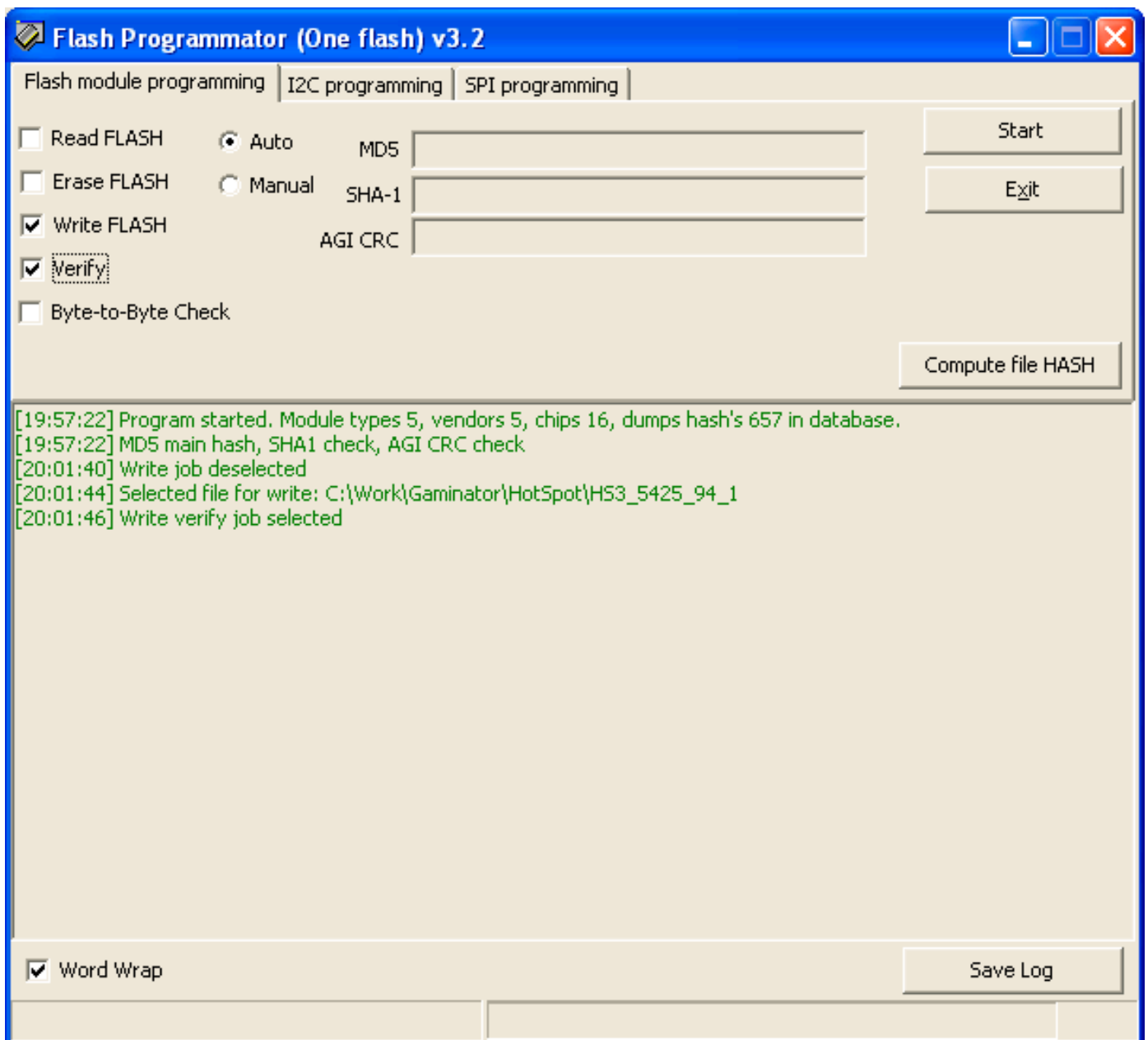
Write Flash

Выбор данного пункта открывает диалоговое окно, в котором необходимо указать, содержимое какого файла будет записано во флеш-модуль



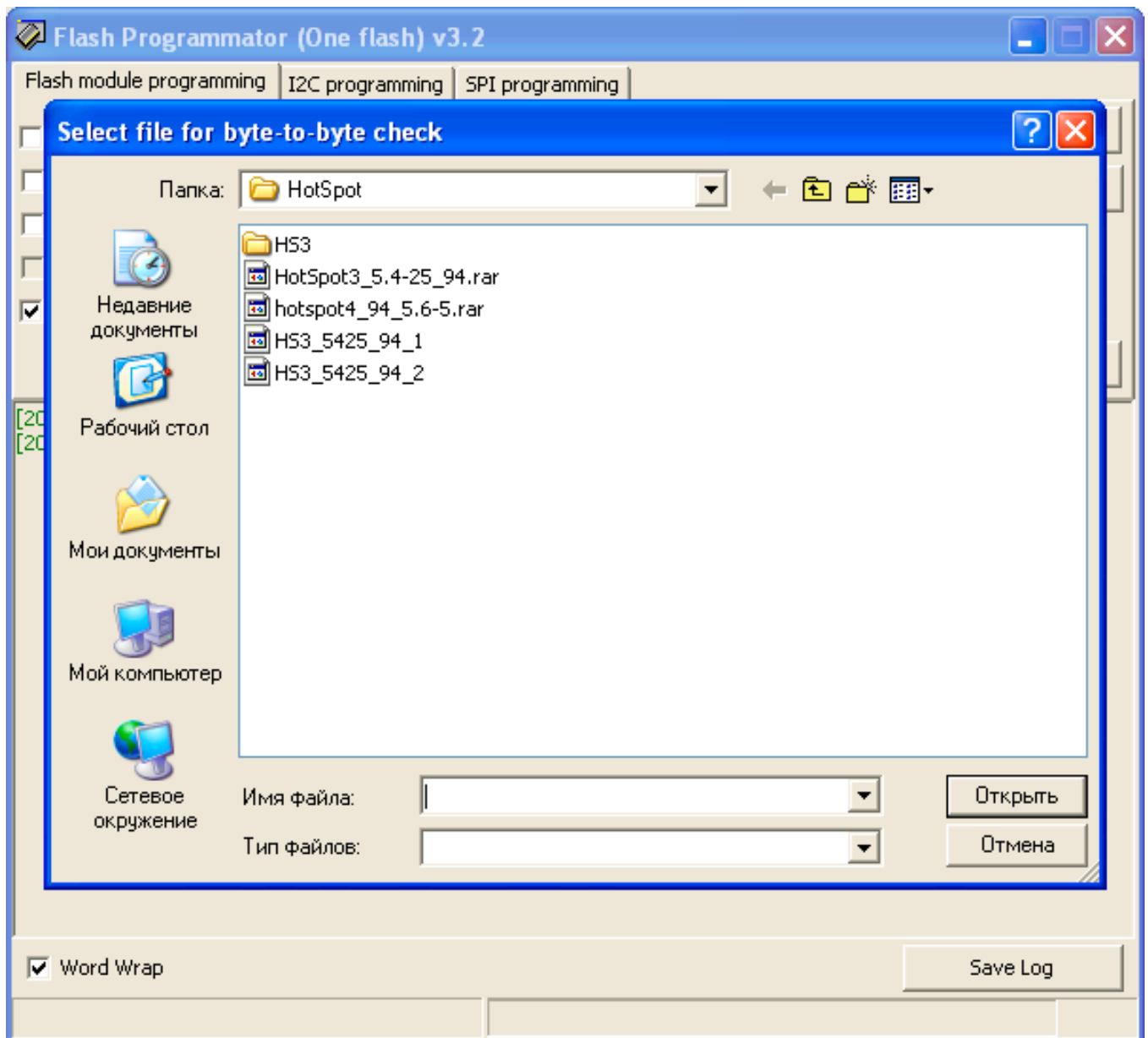
Verify

После выбора пункта **Write Flash**, открывается возможность выбора пункта **Verify**. Если данный режим включен, после записи будет произведена побайтовая проверка содержимого модуля с файлом прошивки.



Byte-to-byte check

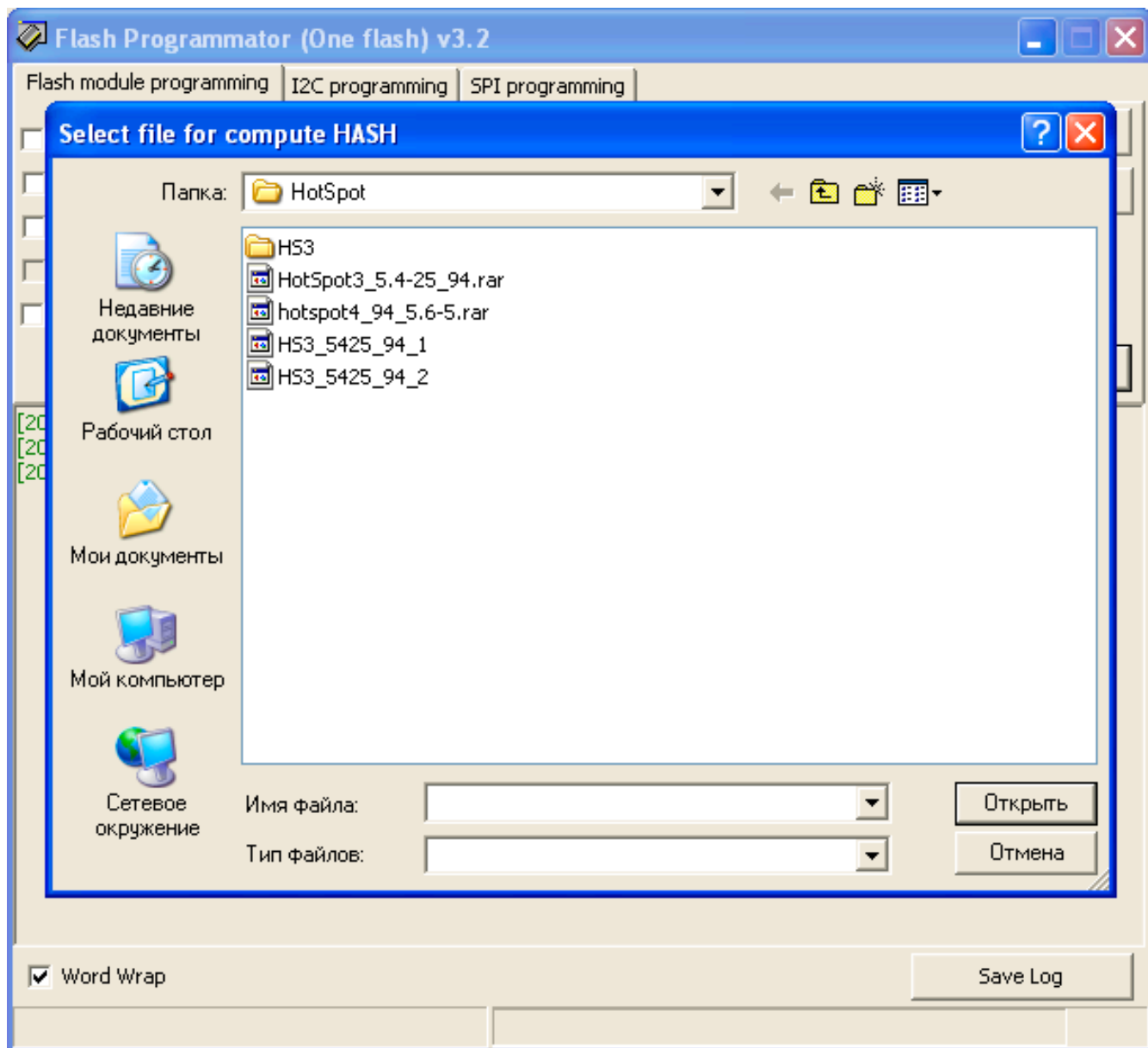
Выбор данного пункта открывает диалоговое окно, в котором необходимо указать, с каким файлом будет произведена побайтовая сверка с содержимым модуля



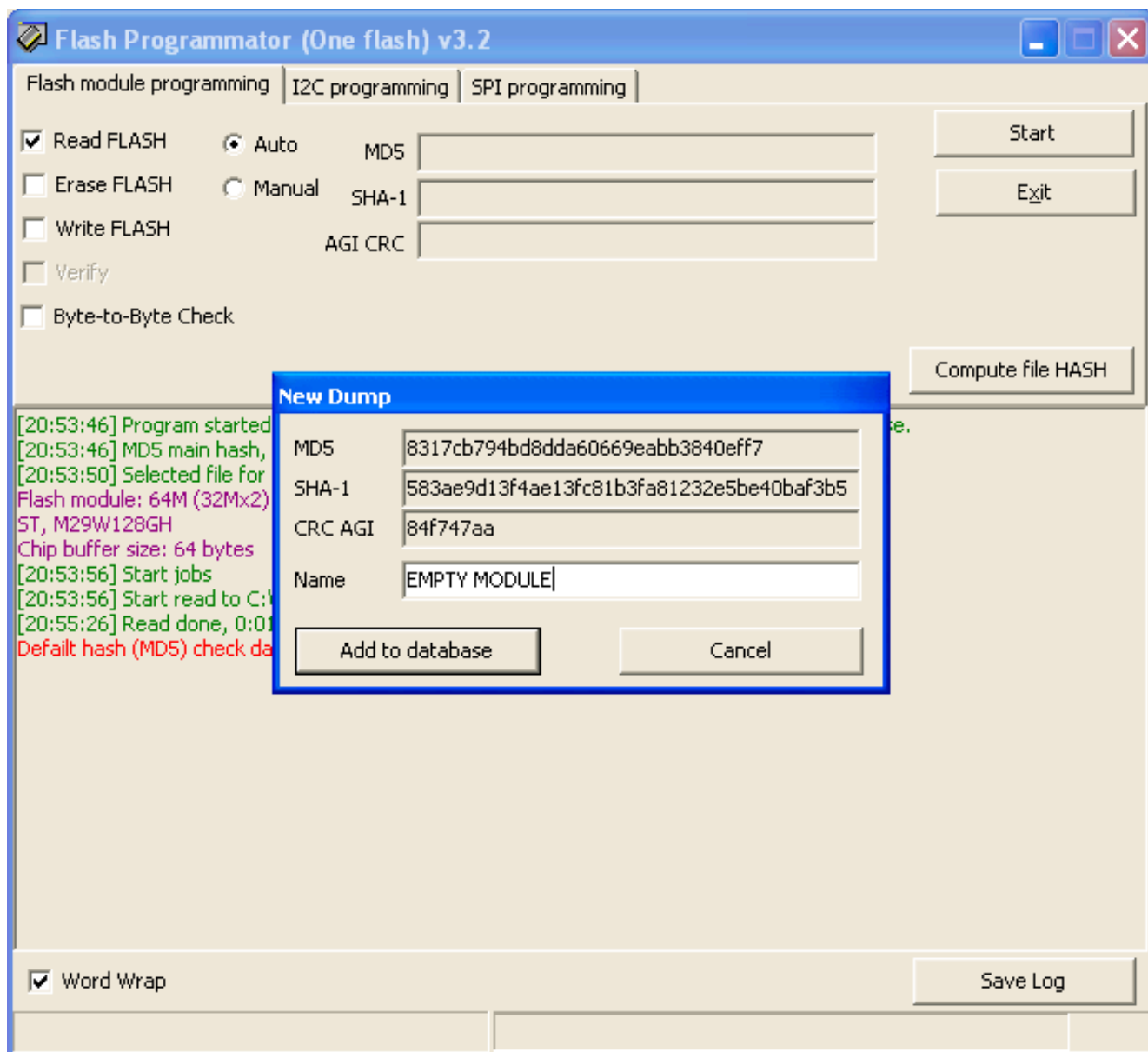
Compute HASH

После нажатия на эту кнопку открывается диалоговое окно, в котором нужно указать файл, для которого рассчитаются контрольные суммы **MD5**, **SHA1** и **AGI CRC**.

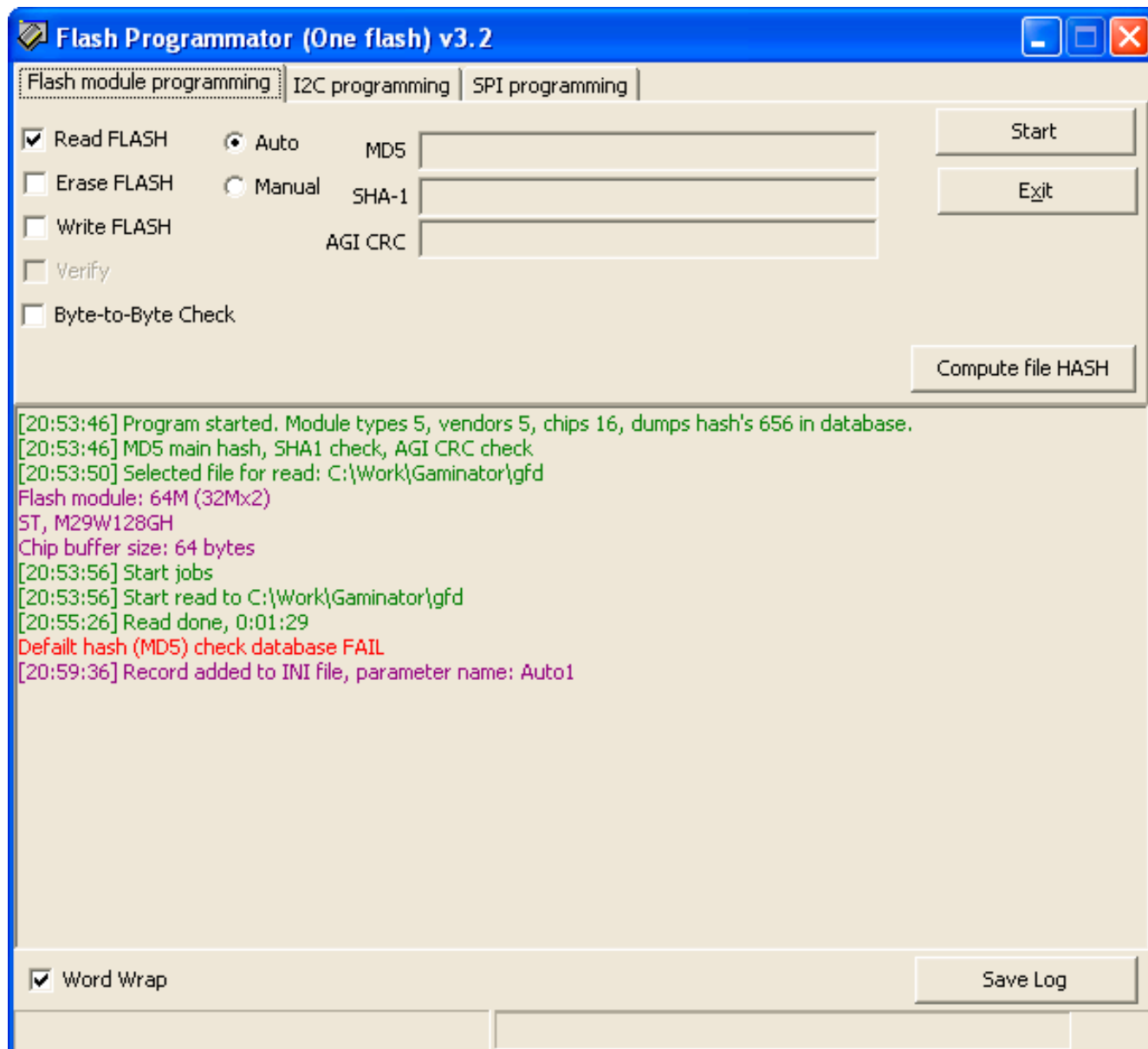
Внимание! Для совершения данной операции программатор должен быть подключен к порту USB!



После чтения модуля или вычисления контрольных сумм файла выполняется проверка с базой контрольных сумм, хранящихся в INI файле. Если совпадений не найдено, Вы можете внести запись в базу, просто указав название



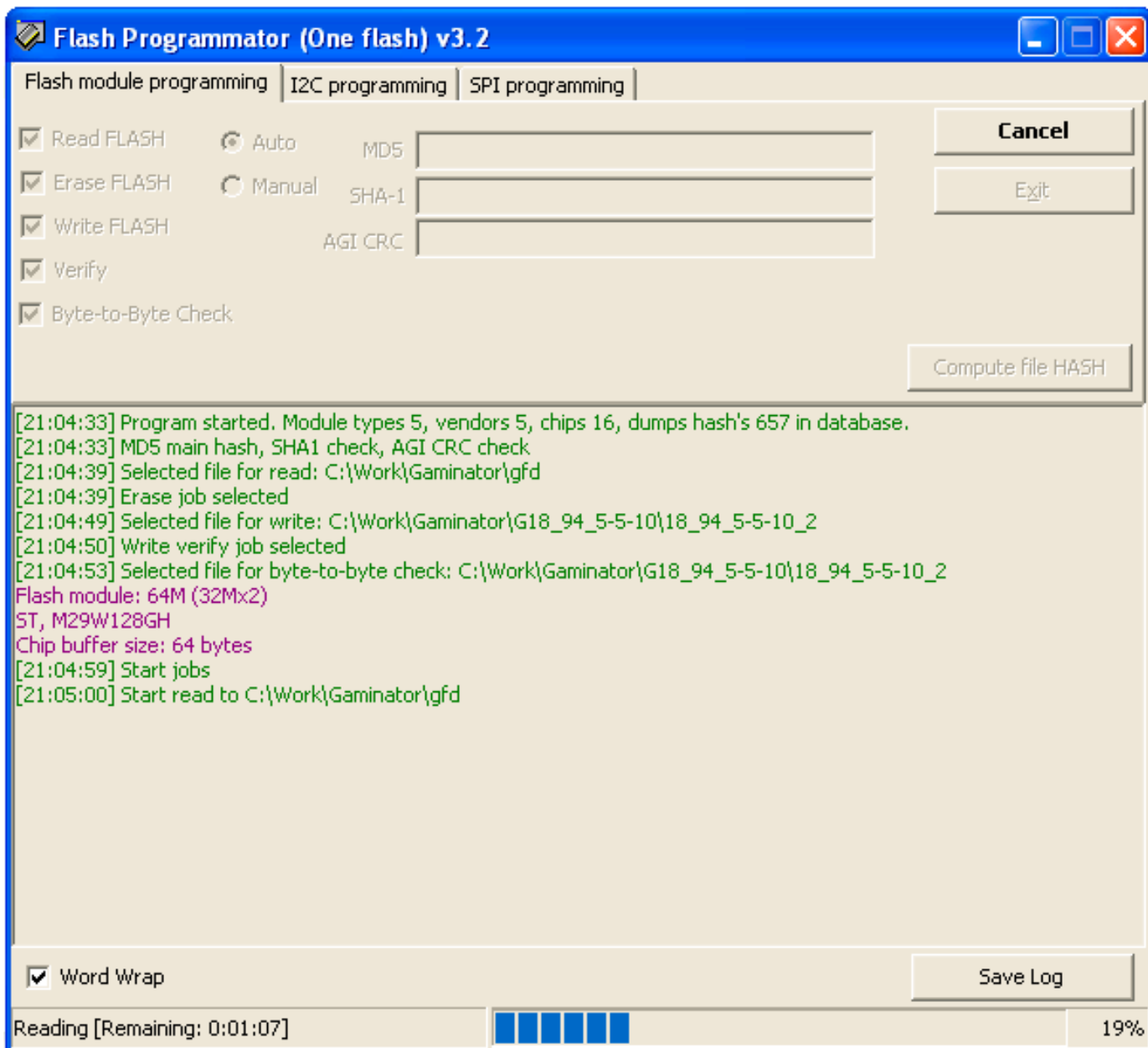
Данные контрольные суммы соответствуют чистому 64-мегабайтному модулю.

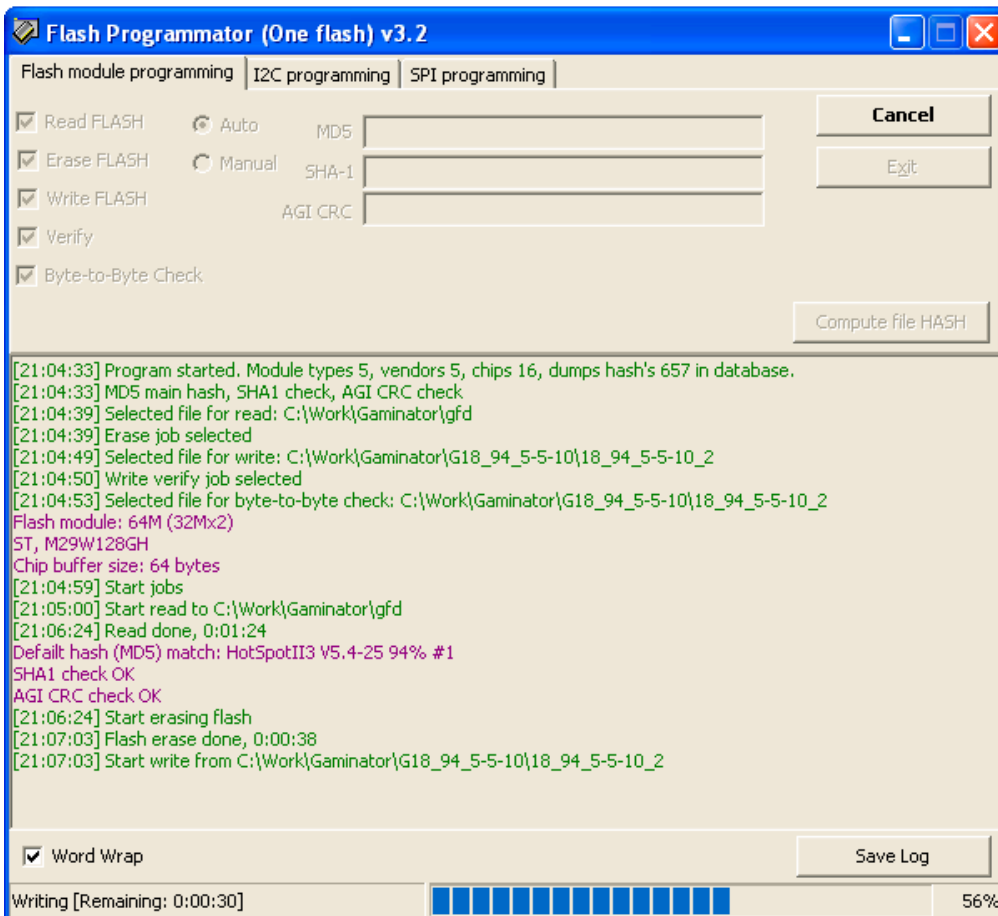
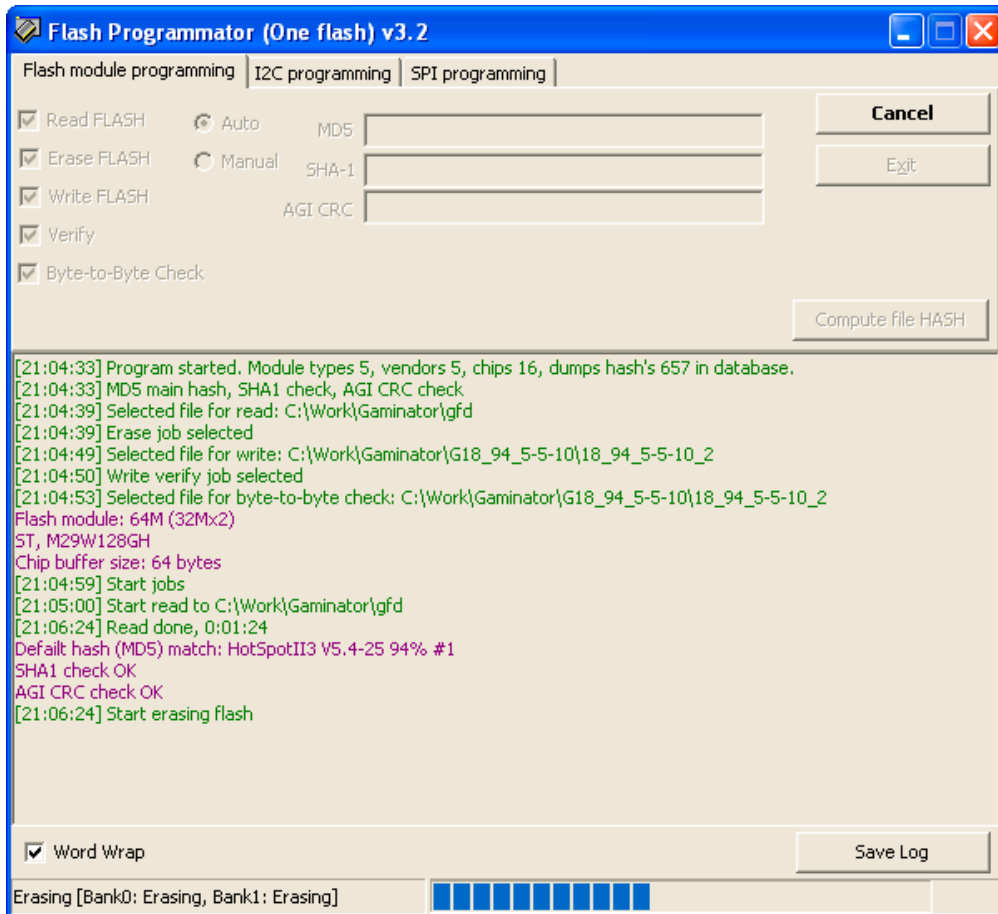


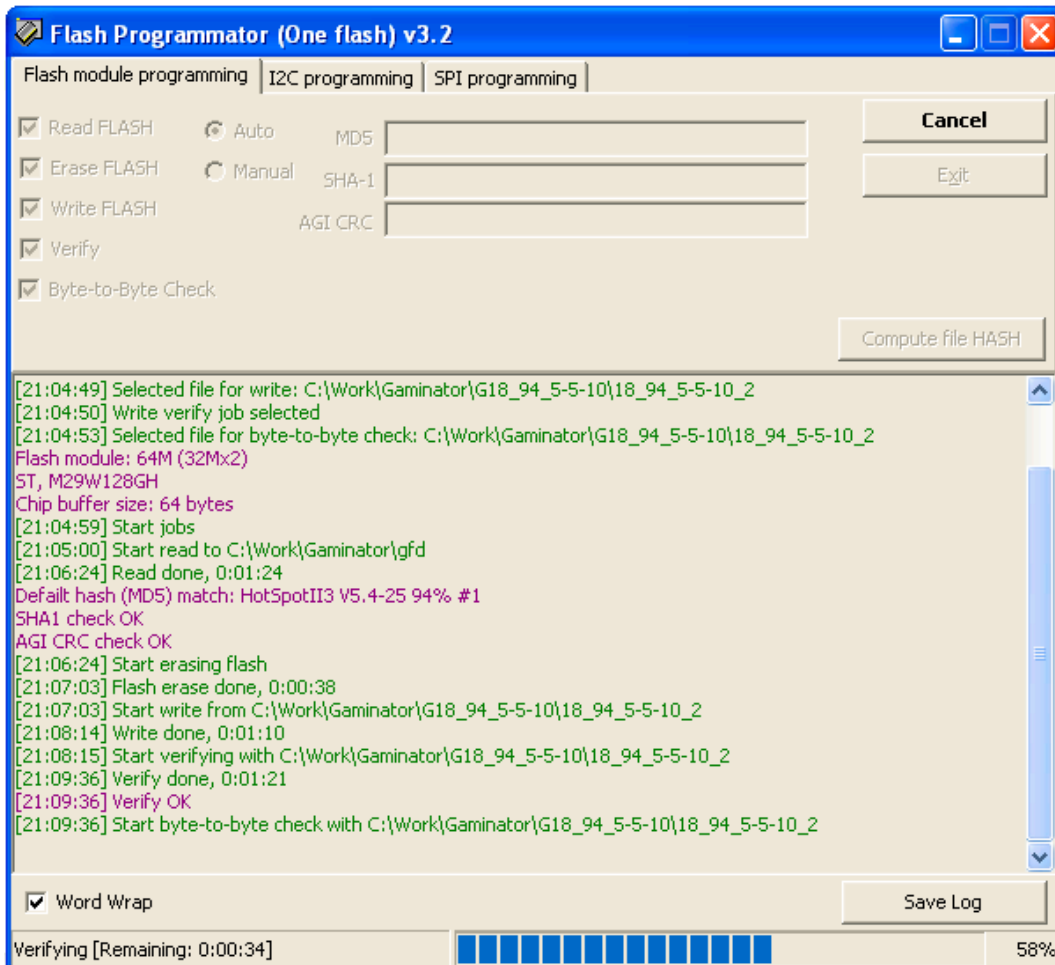
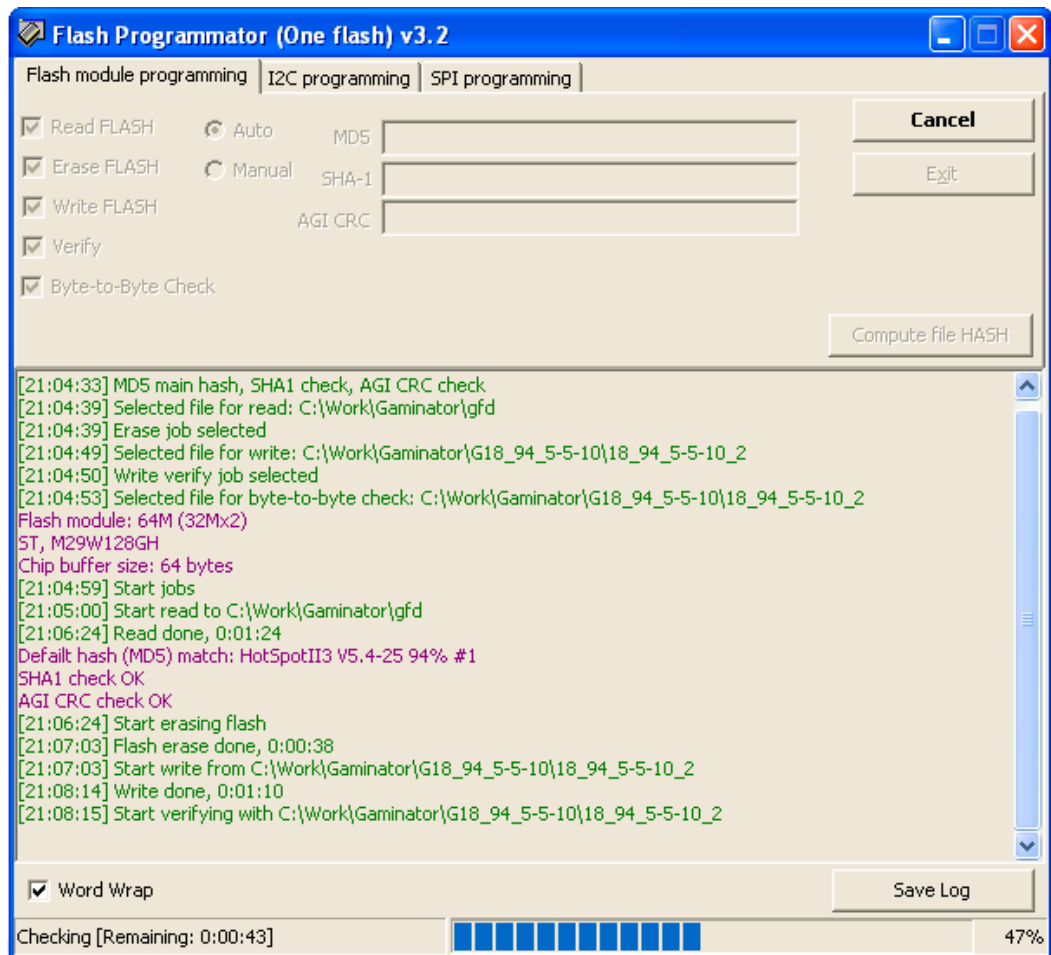
После добавления создается новая запись в секции **[HASH]** INI файла, имя параметра определяется автоматически (в данном случае - **Auto1**)

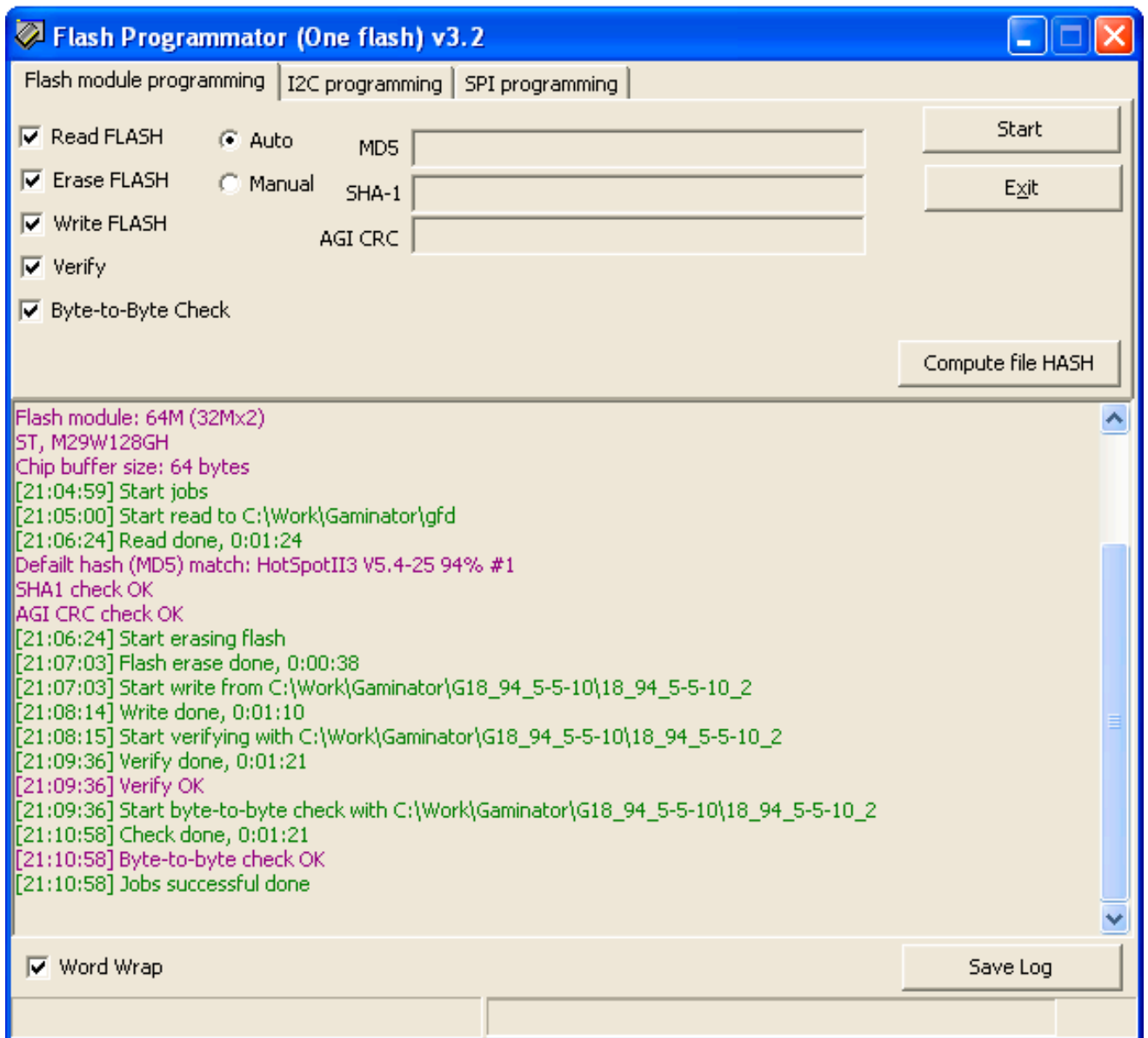
После выбора необходимых заданий, нажимаем кнопку "Start"

Задания выполняются последовательно, одно за другим



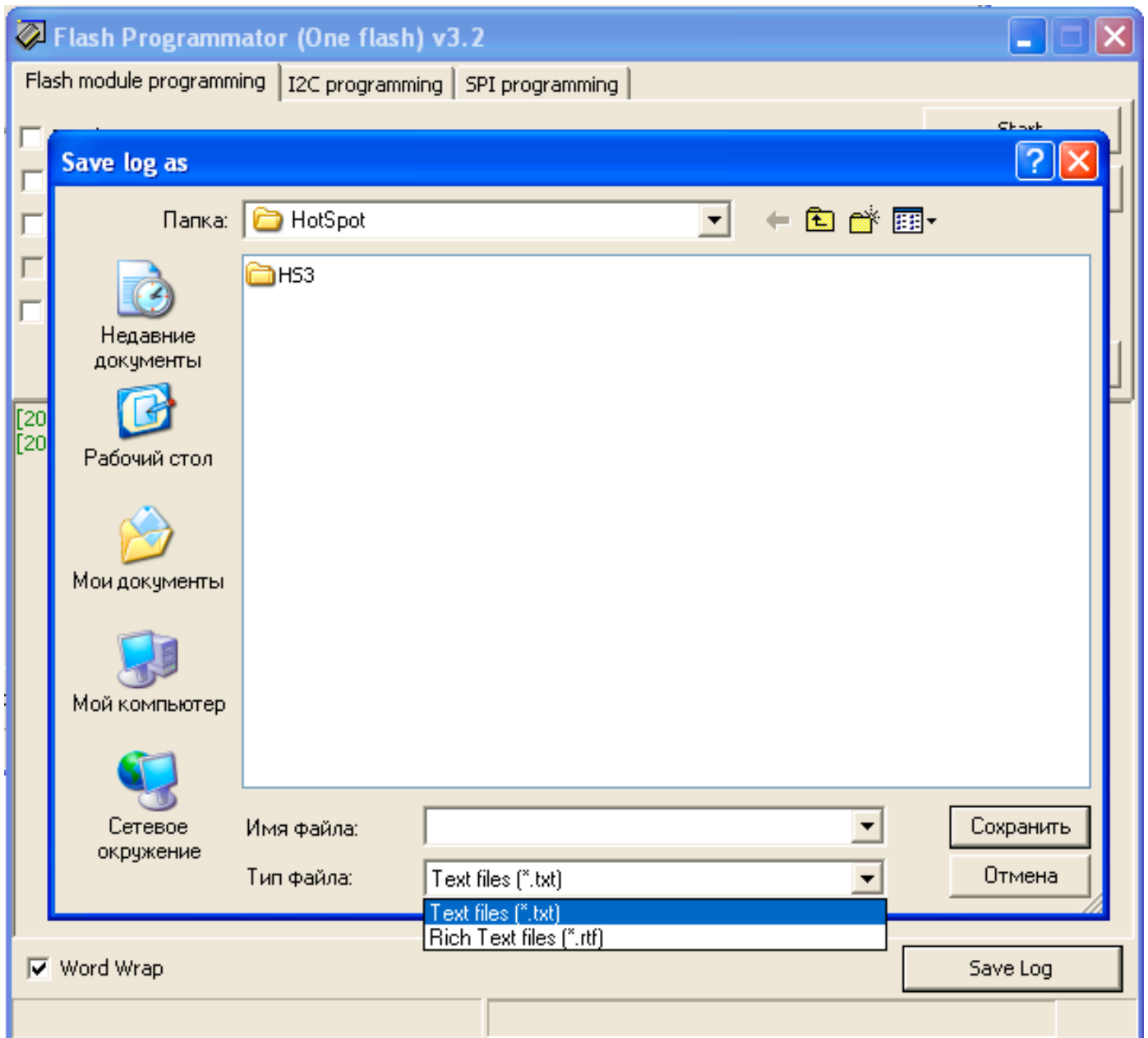






Вы можете сохранить лог, нажав на кнопку Save Log

Поддерживаемые форматы: TXT, RTF



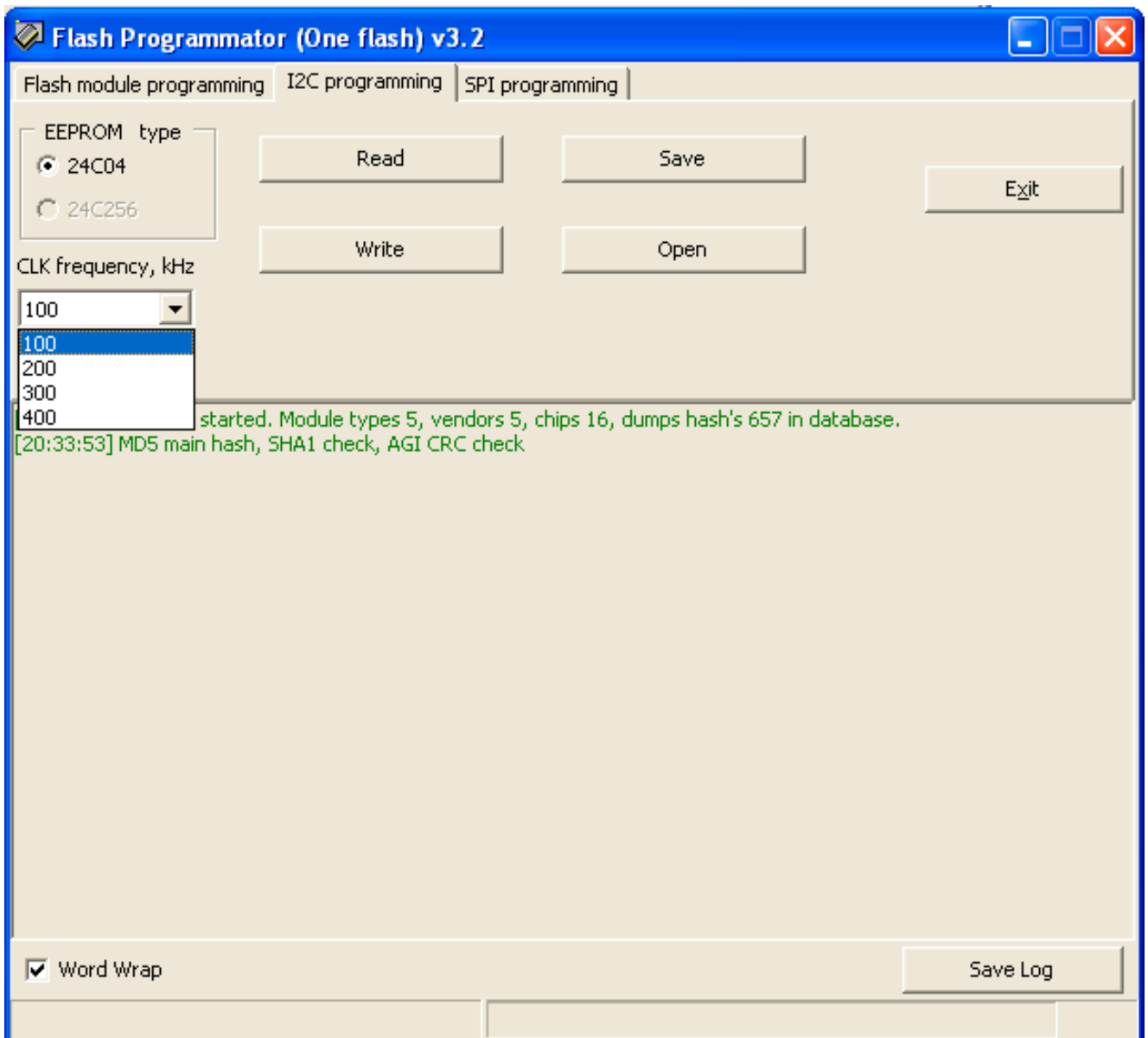
Вкладка I2C programming

Read - чтение содержимого микросхемы в буфер

Write - запись буфера в микросхему

Open - чтение файла в буфер. Поддерживаемые форматы: BIN, Intel HEX

Save - сохранение буфера в файл



Вы можете выбрать частоту записи (100, 200, 300 или 400 КГц)

Вкладка SPI programming

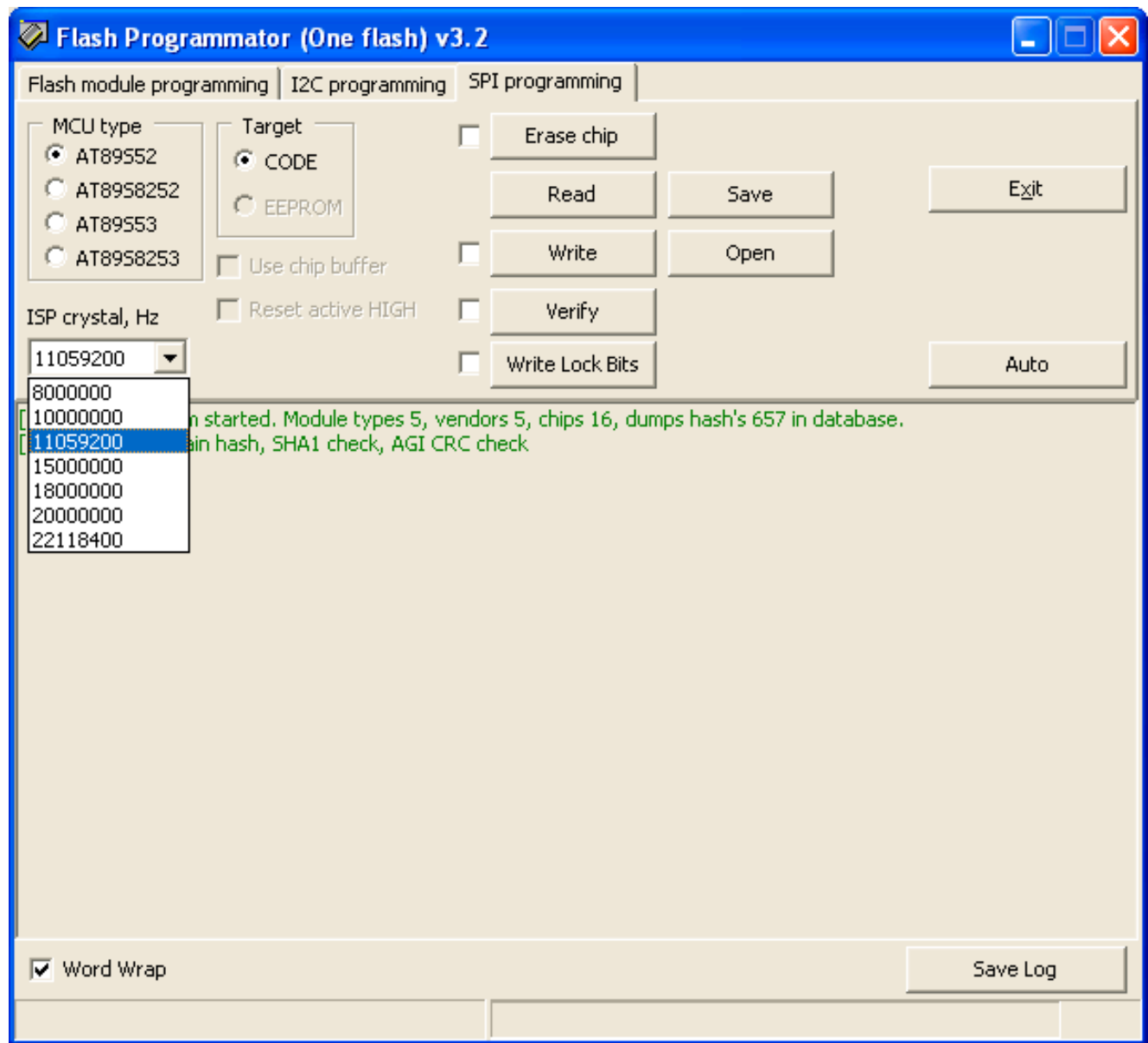
MCU type - выбор типа микроконтроллера

Target - рабочая область микроконтроллера

Write Lock Bits - защита содержимого памяти микроконтроллера от чтения

Verify - проверка записи

Auto - последовательное выполнение отмеченных операций



Вы можете выбрать частоту кварцевого резонатора, используемого в схеме. Заметьте, что иногда понижение реальной частоты позволит получить более стабильное программирование

Конфигурационный INI файл

Секция [Modules]

Содержит описание типов модулей

```
[Modules]
module0 = ID,SIZE,TWO_BANKS
...
```

ID - идентификатор модуля в шестнадцатеричном виде

SIZE - емкость модуля {16|32|64} Мбайт

TWO_BANKS - организация: два банка или один {1|0}

Секция [Vendors]

Описание производителей. Используется в информационных целях

```
[Vendors]
mnf0 = NAME, ID
...
```

NAME - Имя производителя микросхемы

ID - шестнадцатеричный код производителя (ID0 из идентификаторов микросхемы)

Секция [Chips]

Описание используемых микросхем

```
[Chips]
chip0 = NAME,CHIP_ORG,WRITE_METHOD,USING_UBYPASS, ID0, ID1[, ID2, ID3]
...
```

NAME - название микросхемы

CHIP_ORG - организация микросхемы:

0 - 8bit

1 - 16bit

2 - 32bit

WRITE_METHOD - способ записи

0 - word by word

1 - use buffer

2 - use extended buffer

3 - use buffer size from CFI

USING_UBYPASS - использование усеченных команд

0 - не используется

1 - включено

IDx - идентификаторы микросхемы (начиная с Vendor ID). Количество байтов ID - (2 <= IDs <= 4)

Секция [Hash]

База хэш-сумм прошивок

[Hash]

dump = NAME, HASH_MD5, HASH_SHA1, AGI_CRC

...

NAME - название прошивки

HASH_MD5 - 32 символа в 16-ричном виде HASH-суммы MD5

HASH_SHA1 - 40 символов в 16-ричном виде HASH-суммы SHA1

AGI_CRC - 8 символов в 16-ричном виде AGI CRC

Имя значения (в данном случае **dump**) программой не обрабатывается, можно использовать любое имя

Секция [Setup]

Содержит настройки проверки хэшей в базе

[Setup]

MD5_check = M

SHA1_check = 1

AGI_CRC_check = 1

Значения параметров могут быть {M|1|0}

Один из параметров всегда должен быть M (Main)

Метод проверки таков. В базе хэш-сумм ищется сумма, указанная как Main. При удачном поиске сверяются остальные типы хэшей, указанные как "1". Сумма, помеченная как "0", не проверяется

Секция [LED]

Содержит описание цветовой индикации режимов работы

[LED]

parameter = time

mode = COLOR,MODE[,COUNT]

...

Группа настроек частоты вспышек

Parameter	time (default) ⁽¹⁾	Примечание
ShortInterval	10	Время свечения и паузы в режиме быстрого мерцания (Short)
LongInterval	70	Время свечения и паузы в режиме долгого мерцания (Long)
CounterActiveTime	20	Время свечения в счетном режиме (Count)
CounterPassiveTime	20	Время паузы в счетном режиме (Count)
CounterInterval	100	Время паузы между группой вспышек в счетном режиме (Count)

⁽¹⁾ Реальное время рассчитывается как (time*10) мс

Группа настроек свечения для каждого режима

COLOR - цвет. Зеленый (G), красный (R) или желтый (Y)

MODE - режим индикации (L - медленное мерцание, S - короткое мерцание, C - определенное количество вспышек)

COUNT - количество вспышек в режиме "C"

Значения по умолчанию

Mode	Значение
StartReadFlash	G,C,1
FaultReadFlash	R,C,1
StartEraseFlash	G,C,2
FaultEraseFlash	R,C,2
StartWriteFlash	G,C,3
FaultWriteFlash	R,C,3
StartCheckFlash	G,C,4
FaultCheckFlash	R,C,4

StartBTBCheckFlash	G,C,5
FaultBTBCheckFlash	R,C,5
StartReadI2C	Y,S
FaultReadI2C	R,S
StartWriteI2C	Y,S
FaultWriteI2C	R,S
StartReadSPI	Y,S
FaultReadSPI	R,S
StartEraseSPI	Y,S
FaultEraseSPI	R,S
StartWriteSPI	Y,S
FaultWriteSPI	R,S
StartWriteLockbitsSPI	Y,S
FaultWriteLockbitsSPI	R,S

Можно не добавлять все значения в секцию **[LED]**, достаточно добавить только те параметры, для которых необходимо указать новые значения

Контактная информация

По всем вопросам, связанным с эксплуатацией программных и аппаратных средств **toweroff's**, вы можете обратиться:

- на сайте, воспользовавшись формой обратной связи

<http://toweroff.ru/index.php?page=feedback>

- на форуме <http://toweroff.ru/forum/index.php>

Для доступа к некоторым разделам необходима регистрация

На форуме всегда можно скачать свежие базы контрольных сумм прошивок, списки поддерживаемых микросхем и задать интересующие вопросы

- по email toweroff@mail.ru