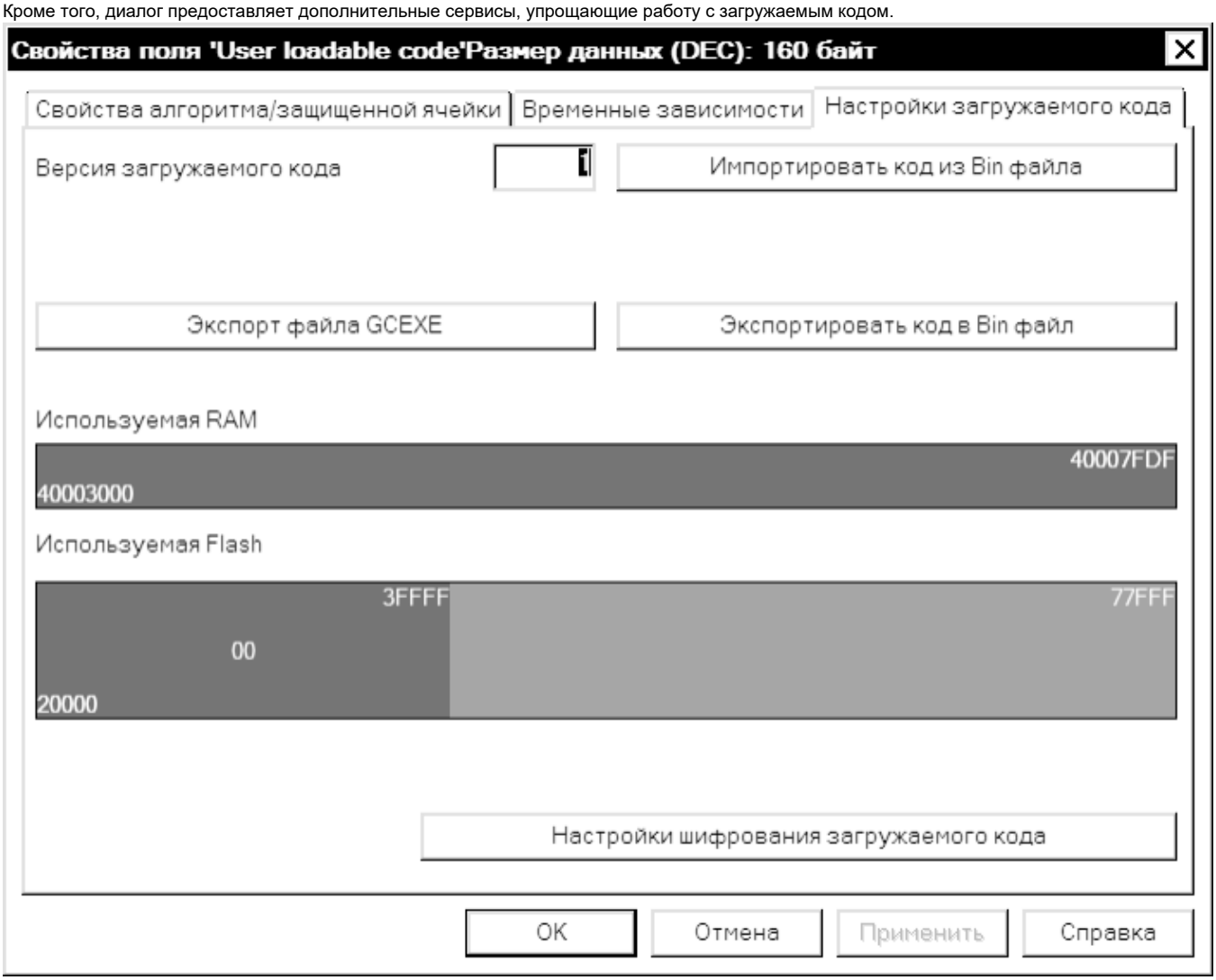
Запись загружаемого кода_

После компиляции кода и получения **Bin-файла** разработчику необходимо передать бинарный код в **GrdUtil.exe** для его обработки и записи в ключ. Для этого служит диалог **Настройки загружаемого кода**.

Диалог позволяет:

- Импортировать предварительно скомпилированный код из Bin-файла
- Преобразовывать импортированный код в формат **GCEXE,** пригодный для записи в ключ **Guardant Code / Code Time**
- Записывать код в **GCEXE-формате** во **Flash-память** ключа или выгружать его во внешний файл.

Кроме того, диалог предоставляет дополнительные сервисы, упрощающие работу с загружаемым кодом.



- Импорт загружаемого кода из Bin-файла
- Преобразование бинарного кода в формат GCEXE
- Настройки шифрования загружаемого кода
- Экспорт файла GCEXE. Обновление кода у пользователя
- Запись загружаемого кода в ключ

Импорт загружаемого кода из Bin-файла

По нажатию кнопки **Импортировать код из Bin-файла**, находящейся в правой верхней части страницы **Настройки загружаемого кода**, появляется диалог выбора Bin-файла из нужного проекта.

При импорте **GrdUtil.exe** считывает из файла **имя_проекта.bmap** настройки, описывающие использование памяти ключа загружаемым кодом. После этого в диалоге отображаются:

Индикатор состояния	**Назначение**
Используемая RAM	Индикация выделенной для загружаемого кода оперативной памяти ключа, ее начального и конечного адреса
Используемая Flash	Индикация выделенной для загружаемого кода Flash-памяти ключа, ее начального и конечного адреса, а также номера ячейки, хранящей дескриптор загружаемого кода

Свободная память обозначается зеленым цветом, используемая – синим. Адресация дается в шестнадцатеричном формате.

Преобразование бинарного кода в формат GCEXE

По соображениям конфиденциальности загружаемый код не должен передаваться «наружу» в открытом виде.

Поэтому в **GrdUtil.exe** реализована эффективная схема подготовки кода для записи в электронный ключ и безопасной передачи обновлений загружаемого кода конечным пользователям.

GrdUtil.exe автоматически преобразует бинарный код в файл формата **GCEXE,** содержащий:

- Зашифрованный на AES исходный код
- Зашифрованный на *открытом* ключе ***ECC160 №#1*** сеансовый ключ AES, использовавшийся ранее для шифрования кода
- ЭЦП файла, полученную на закрытом ключе **ECC160 №#2**

При этом в дескрипторе (ячейке) загружаемого кода хранится «ответная часть» ключей ECC, используемых при преобразовании бинарного кода:

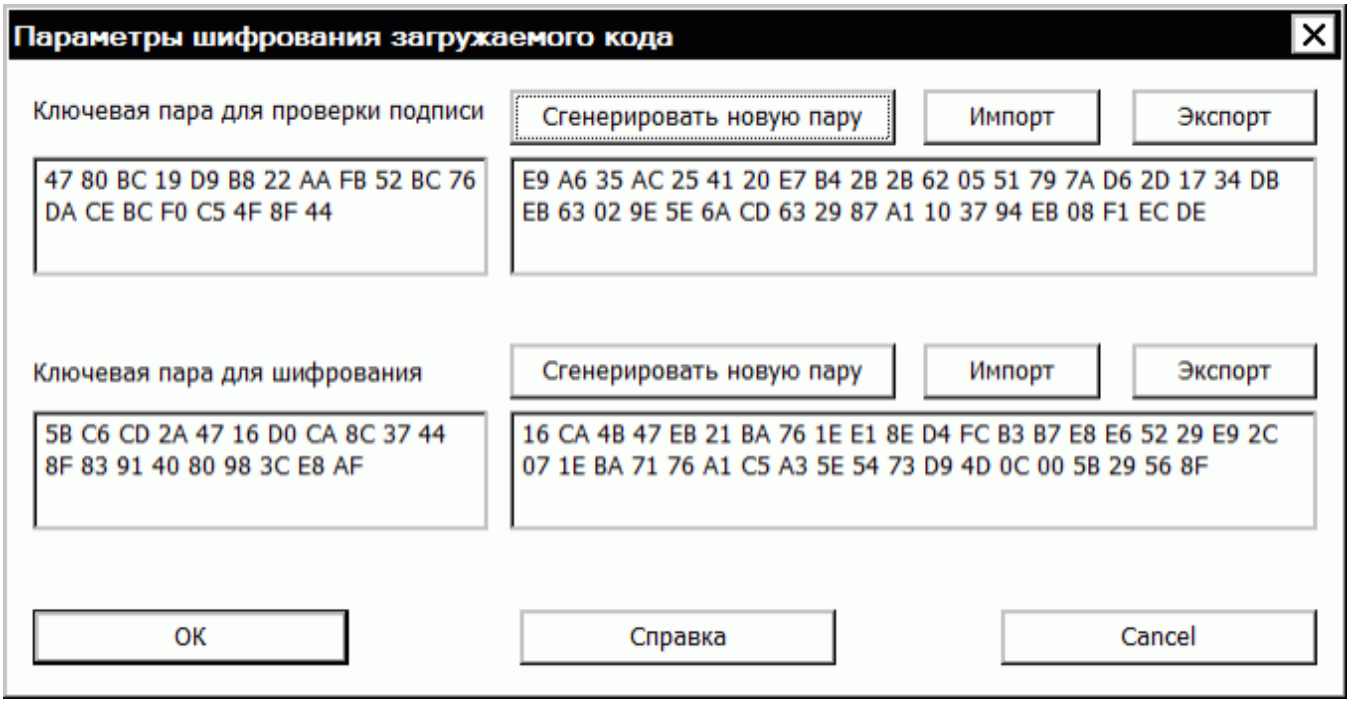
- *Закрытый* ключ ***ECC160 №#1*** для шифрования
- Открытый ключ **ECC160 №#2** для ЭЦП

Что позволяет электронному ключу при обращении к загруженному коду успешно его проверять, расшифровывать и выполнять.

Важно!
Преобразование кода в формат **GCEXE** производится утилитой **GrdUtil.exe автоматически** при записи образа в ключ (либо нажатии на кнопку **Экспортировать GCEXE**), не требуя от разработчиков никаких действий для ее реализации, кроме настройки ключевых пар ECC160.

Настройки шифрования загружаемого кода

По нажатию кнопки **Настройки шифрования загружаемого кода** появляется диалог для работы с ключевыми парами:



Диалог предназначен для генерации, импорта и экспорта ключевых пар асимметричного алгоритма **ECC160,** которые используются при преобразовании бинарного кода в формат **GCEXE** (см. предыдущий пункт).

В верхней части диалога отображаются закрытый (слева) и открытый (справа) ключи **ECC160 №#2** для цифровой подписи зашифрованного кода.

В нижней части диалога находится ключевая пара (закрытый ключ – слева, открытый – справа) ***ECC160 №#1*** для шифрования бинарного кода.

Кроме того, диалог дополнен кнопками, позволяющими генерировать новые ключевые пары, экспортировать их во внешний файл для использования в приложении и импортировать ключевые файлы из других проектов.

Экспорт файла GCEXE. Обновление кода у пользователя

Экспорт **GCEXE** во внешний файл может потребоваться в случае, когда разработчику необходимо обновить загружаемый код в электронном ключе, находящемся у конечного пользователя.

В такой ситуации разработчику следует придерживаться следующей схемы действий:

1. На этапе разработки приложения должен быть предусмотрен механизм обновления загружаемого кода из приложения. Такой механизм реализуется при помощи функции Guardant API **GrdCodeLoad.**
2. После внесения необходимых изменений новая версия загружаемого кода компилируется в бинарный файл, который импортируется в **GrdUtil.exe** (см. ***Импорт загружаемого кода из Bin-файла***).
3. Для правильной работы обновленного загружаемого кода в удаленном ключе должны использоваться те же ключевые пары, которые применялись при программировании ключа в первый раз (см. ***Настройки шифрования загружаемого кода)***.
4. При нажатии на кнопку **[Экспортировать GCEXE]** происходит формирование и выгрузка **GCEXE** во внешний файл.
5. Если необходимо сделать обновление кода зависимым от ключа (к примеру, при подготовке платных обновлений), то следует указать десятичный ID ключа конечного пользователя в диалоге, который возникает по нажатию кнопки **[Экспортировать GCEXE]**.
6. Сохраненный в формате **GCEXE** загружаемый код передается конечному пользователю, который производит обновление содержимого ключа способом, предусмотренным разработчиком на первом шаге.

Запись загружаемого кода в ключ

После выполнения настроек загружаемого кода остается завершить диалог и выполнить команду меню **Ключ | (Операции с ключом) Записать образ в ключ**. При этом будут сформированы и записаны в ключ:

1. Прошивка, содержащая дескриптор загружаемого кода (в числе прочих полей), – в **EEPROM-память.**
2. Загружаемый код в формате **GCEXE** – во **Flash-память** ключа.