

КриптоПро PDF

Версия 1.6

Руководство по автоматизации создания и проверки
электронных подписей

АННОТАЦИЯ

Настоящий документ содержит описание различных способов автоматизации создания и проверки электронных подписей с помощью Adobe JavaScript и iText. В документе представлены инструкции по использованию мастера пакетной работы с электронной подписью и утилиты командной строки, которые входят в состав «КриптоПро PDF».

Информация о разработчике «КриптоПро PDF»:

ООО «Крипто-Про»

127 018, Москва, а/я Крипто-Про

Телефон: +7 495 995-48-20

Факс: +7 495 995-48-20

<http://www.cryptopro.ru>

E-mail: info@cryptopro.ru

СОДЕРЖАНИЕ

1. Пакетная обработка документов с использованием КриптоПро PDF	4
1.1. Мастер пакетной работы с электронной подписью	5
1.1.1. Создание электронных подписей	5
1.1.2. Проверка электронных подписей	14
1.2. Программа командной строки crrpdfutil.exe	20
1.2.1. Создание электронных подписей	20
1.2.2. Проверка электронных подписей	21
2. Автоматизация создания и проверки подписи	23
2.1. Автоматизация с использованием Adobe JavaScript	23
2.1.1. Пример кода для формирования электронной подписи	24
2.1.2. Пример кода для проверки электронной подписи	24
2.1.3. Отладка и выполнение кода JavaScript в программе Adobe Acrobat	25
2.2. Автоматизация с использованием iText	29
2.2.1. Разработка на языке Java	29
2.2.2. Разработка на платформе Microsoft .NET Framework	29
2.3. Автоматизация с использованием JavaScript и OLE	30
3. Настройка оформления видимой подписи	33
3.1. Использование предустановленного оформления подписи	33
3.2. Создание настраиваемого оформления подписи	34
4. Перечень сокращений	38
5. Перечень рисунков	39
6. Перечень таблиц	40

1. Пакетная обработка документов с использованием КриптоПро PDF

«КриптоПро PDF» предоставляет следующие возможности для работы с пакетом документов PDF:

- создание электронных подписей для всех документов PDF, входящих в пакет;
- проверку электронных подписей во всех документах PDF, входящих в пакет.

Создание и проверка ЭП в пакете документов PDF могут быть произведены при помощи мастера пакетной работы с ЭП (cppdfwizard.exe) или при помощи утилиты командной строки (cppdfutil.exe).



Для использования мастера пакетной работы с ЭП и утилиты командной строки требуется установить программу Adobe Acrobat Standard или Adobe Acrobat Pro.

1.1. Мастер пакетной работы с электронной подписью

1.1.1. Создание электронных подписей

Шаг 1. Начало работы мастера

Для создания ЭП в пакете документов PDF запустите мастер пакетной работы с электронной подписью (**Пуск** ⇒ **Все программы** ⇒ **КРИПТО-ПРО** ⇒ **Пакетная подпись документов PDF**). На экране появится окно мастера (см. *Рисунок 1*). Для пропуска этого шага при следующем запуске мастера, необходимо поставить флаг **Не показывать эту страницу**.

Нажмите на кнопку **Далее**, чтобы продолжить работу мастера.

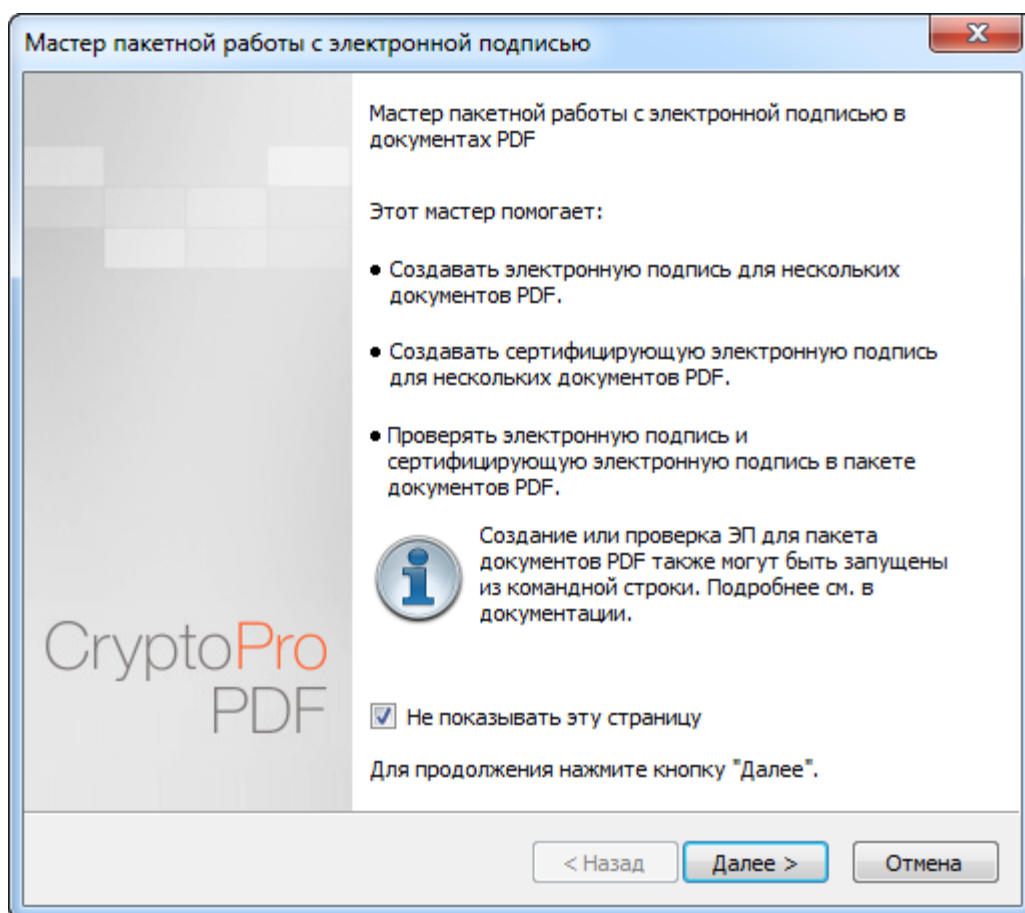


Рисунок 1. Мастер пакетной работы с электронной подписью

Шаг 2. Выбор действия с пакетом документов

В этом окне мастера выберите опцию **Создание электронной подписи в документах** (см. *Рисунок 2*) и нажмите на кнопку **Далее**.

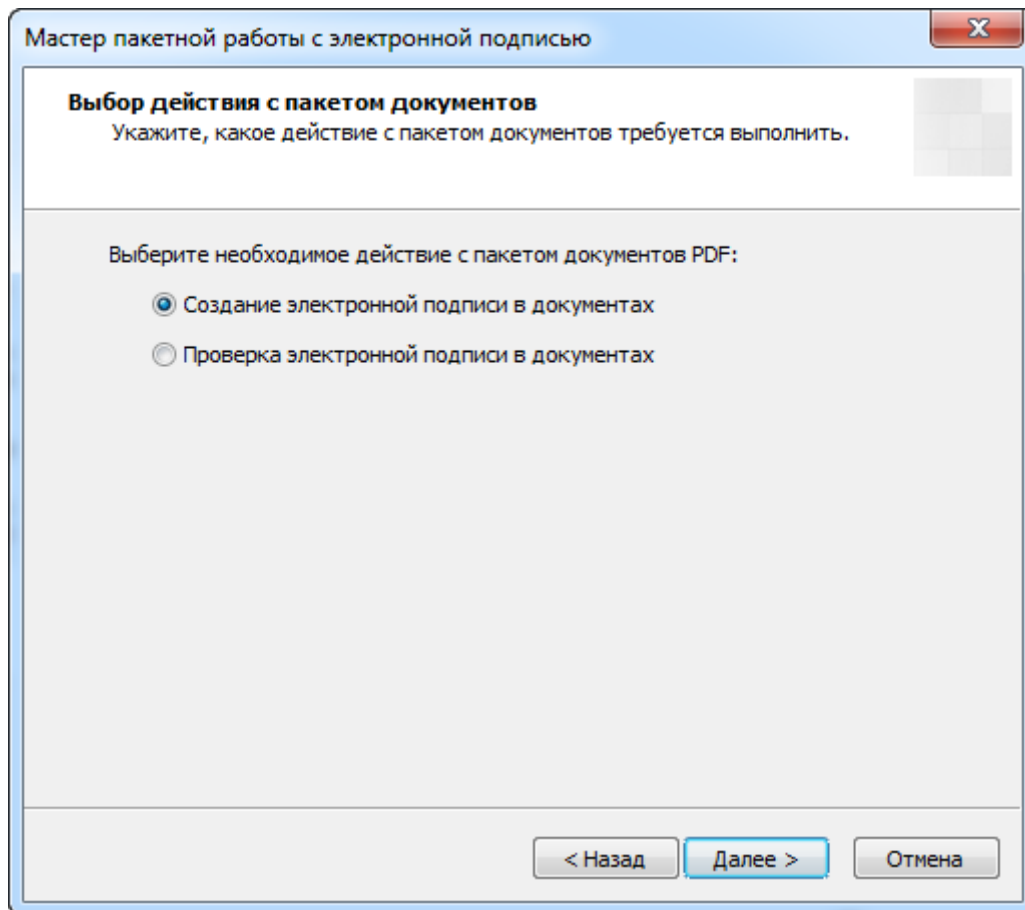


Рисунок 2. Выбор действия с пакетом документов

Шаг 3. Выбор директорий при пакетном создании электронных подписей

На этом шаге мастер отобразит диалоговое окно (см. *Рисунок 3*), в котором необходимо указать директории для работы с пакетом документов PDF:

- **Директория с документами на подпись.** Исходная директория, в которой должны содержаться документы PDF, которые необходимо подписать. Все документы в этой директории останутся неизменными.
- **Директория для подписанных документов.** Директория, в которую будут помещены подписанные копии всех документов PDF из директории с документами на подпись.

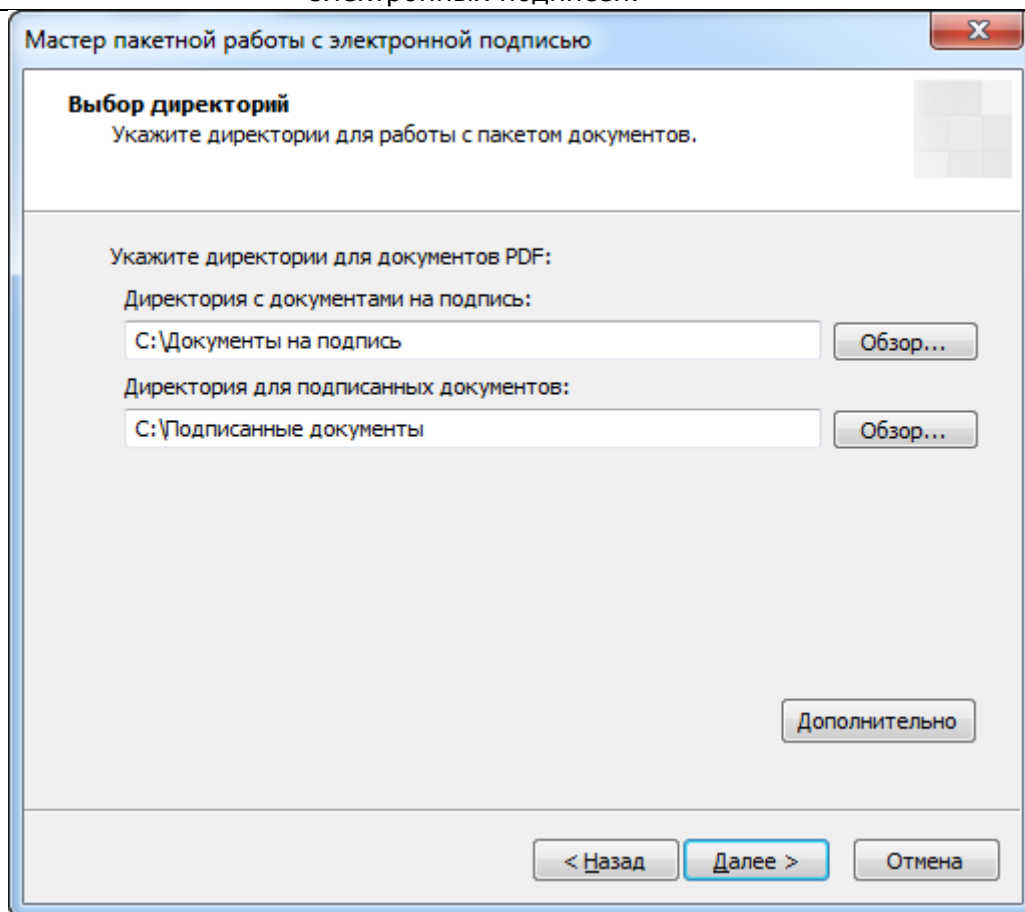


Рисунок 3. Выбор директорий при пакетном создании ЭП

При нажатии на кнопку **Дополнительно** появится окно с дополнительными настройками (см. Рисунок 4).

В этом окне можно указать следующие настройки:

- **Директория для журнальных файлов.** В данную директорию будет помещен файл с отчетом о ходе подписи пакета документов PDF.
- **Директория для документов, которые не удалось подписать.** В данную директорию будут помещены копии всех документов PDF из директории с документами на подпись, которые не были подписаны.
- **Перезаписывать файлы с одинаковыми именами.** При создании файла в директории для подписанных документов или в директории для документов, которые не удалось подписать, уже может существовать файл с таким же именем. Если данная опция включена, то при совпадении имен файл будет перезаписан на новый. Если данная опция отключена, то при совпадении имен процесс создания подписи документе будет прерван.
- **Продолжать работу при возникновении ошибок.** Если данная опция отключена, то при возникновении ошибки программа прекратит процесс создания ЭП и оставшиеся документы PDF подписаны не будут.

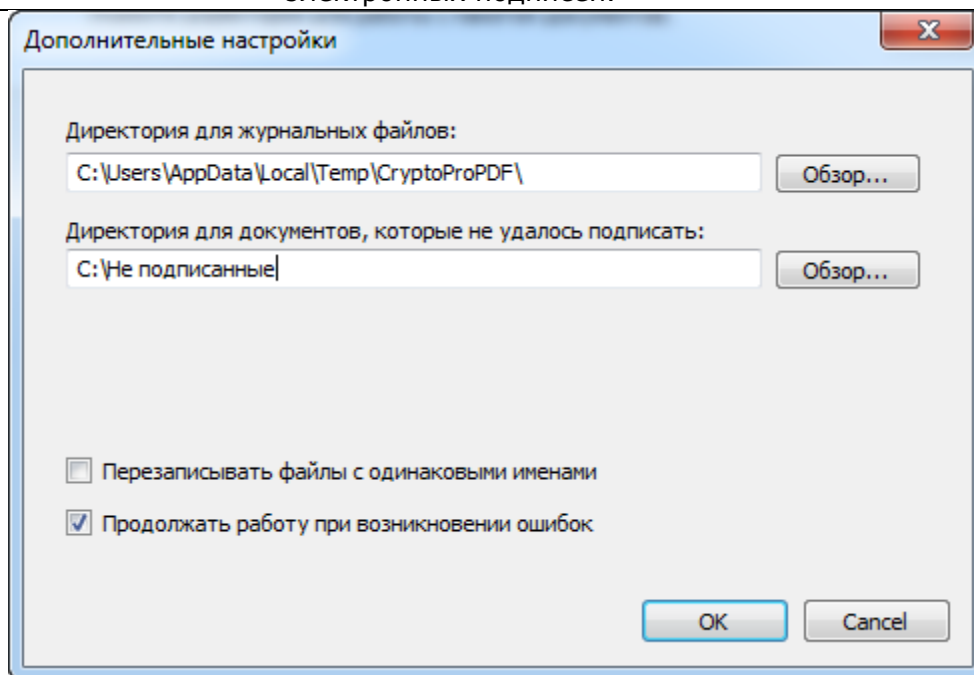


Рисунок 4. Дополнительные настройки

Нажмите на кнопку **Далее**, чтобы продолжить работу мастера.

Шаг 4. Выбор сертификата подписи

На данном этапе мастер отобразит окно выбора сертификата подписи (см. *Рисунок 5*). Сертификат, на котором следует создать ЭП, можно выбрать из хранилища «Личные» текущего пользователя при нажатии на кнопку **Выбрать из хранилища** или из контейнера при нажатии на кнопку **Выбрать из контейнера**.

Просмотреть выбранный сертификат можно при помощи кнопки **Просмотр**.

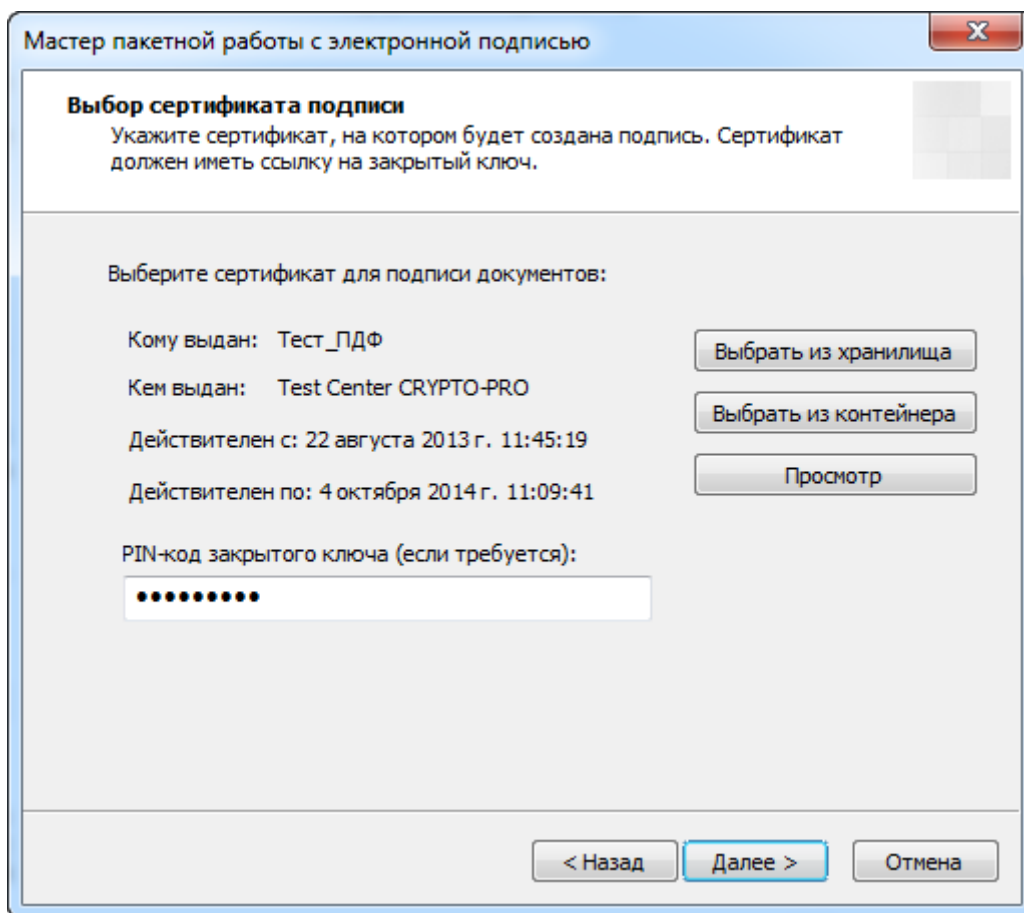


Рисунок 5. Выбор сертификата подписи

Шаг 5. Выбор типа подписи.

На этом шаге мастера (см. *Рисунок 6*) необходимо указать тип создаваемой подписи: обычная или сертифицирующая.

Для указания, должна ли подпись соответствовать усовершенствованному формату CAAdES-X Long Type 1, необходимо выбрать опцию **Использовать усовершенствованную электронную подпись**. В противном случае будет создана подпись, соответствующая формату CAAdES BES.

При создании усовершенствованной подписи должен быть указан адрес службы штампов времени в поле **Служба штампов времени**. Если адрес службы штампов времени не задан, то он может быть взят из групповой политики.

Также необходимо выбрать, какие действия с документом будут разрешены после подписания. Если создается обычная подпись, то можно заблокировать документ после подписания. Если создается сертифицирующая подпись, то можно выбрать одну из настроек:

- разрешить заполнение форм и использование цифровых подписей;
- разрешить комментарии, заполнение форм и использование цифровых подписей;
- запретить вносить изменения.

Мастер пакетной работы с электронной подписью

Тип подписи
Выберите тип подписи.

Тип подписи

Обычная подпись

Сертифицирующая подпись

Служба штампов времени

Использовать усовершенствованную электронную подпись

Укажите адрес службы штампов времени:

Действия, разрешенные после сертификации

Комментарии, заполнение полей форм и использование цифровых подписей

Заполнение полей форм и использование цифровых подписей

Изменения запрещены

< Назад Далее > Отмена

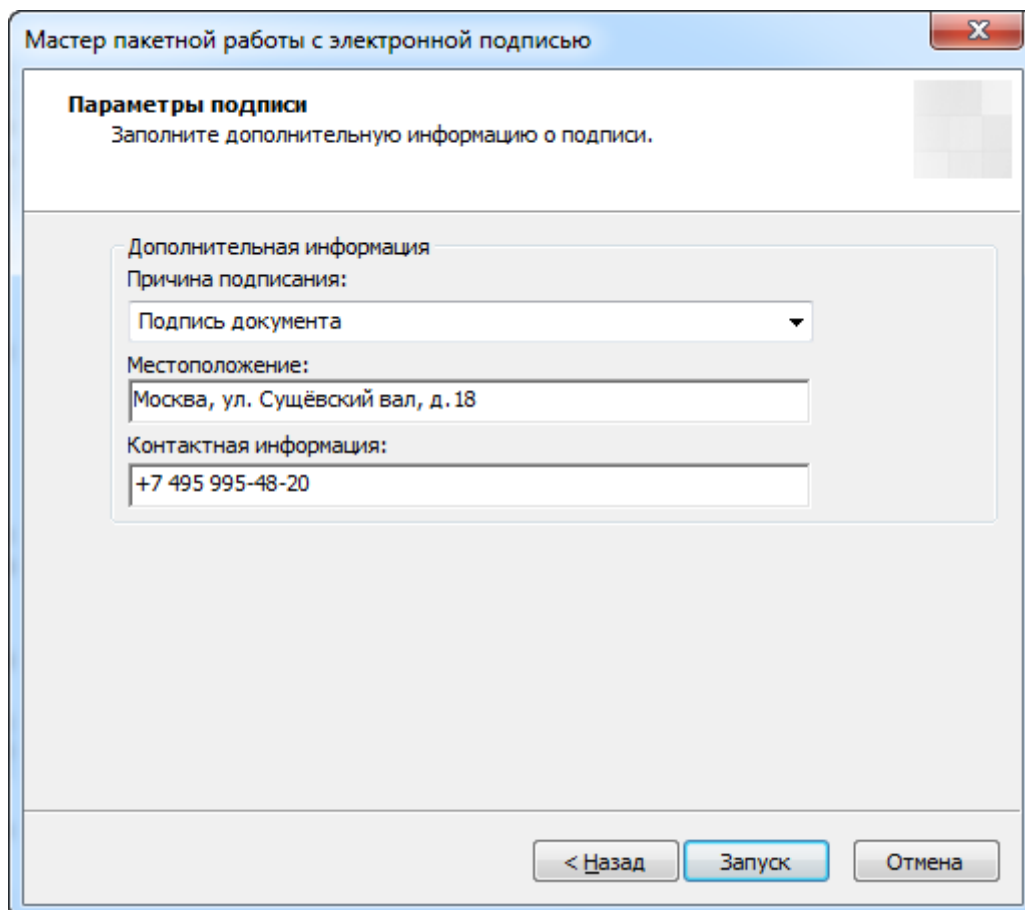
Рисунок 6. Выбор типа подписи

Шаг 6. Ввод параметров подписи

На данном шаге мастера можно указать дополнительную информацию о подписи:

- причину подписания (можно выбрать из списка или ввести вручную);
- местоположение;
- контактную информацию.

Для запуска процесса создания ЭП в выбранных документах PDF, нажмите на кнопку **Запуск**.



The image shows a software dialog box titled "Мастер пакетной работы с электронной подписью" (Batch work master with electronic signature). The current step is "Параметры подписи" (Signature parameters), with the instruction "Заполните дополнительную информацию о подписи." (Fill in additional information about the signature.).

The dialog contains a section titled "Дополнительная информация" (Additional information) with the following fields:

- "Причина подписания:" (Reason for signing): A dropdown menu with "Подпись документа" (Document signature) selected.
- "Местоположение:" (Location): A text field containing "Москва, ул. Суцёвский вал, д. 18" (Moscow, Sucevskiy Val, d. 18).
- "Контактная информация:" (Contact information): A text field containing "+7 495 995-48-20".

At the bottom of the dialog, there are three buttons: "< Назад" (Back), "Запуск" (Start), and "Отмена" (Cancel).

Рисунок 7. Параметры подписи

Шаг 7. Создание электронной подписи

На данном этапе мастер начнет создавать ЭП в выбранных документах, отображая ход процесса подписания в таблице (см. *Рисунок 8*).

Если опция **Продолжать работу при возникновении ошибок** (см. *Рисунок 4*) отключена, то при возникновении ошибки программа прекратит процесс создания ЭП и оставшиеся документы PDF не будут подписаны.

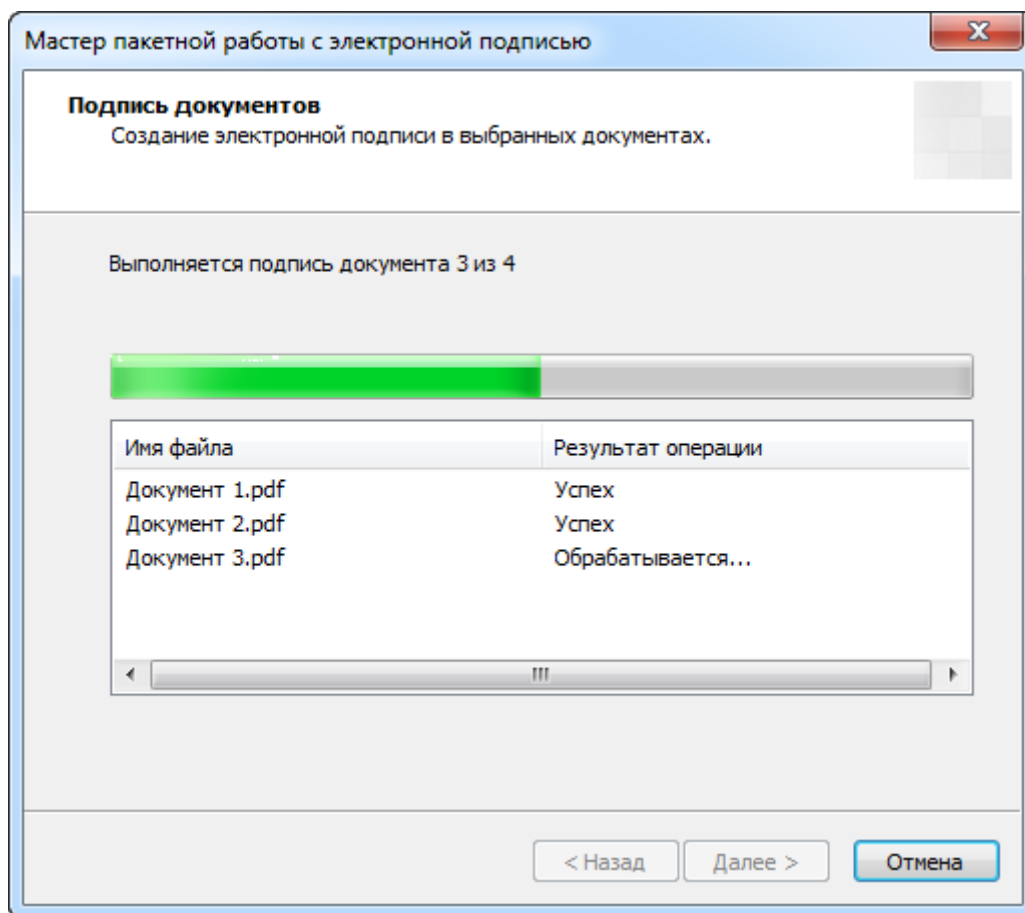


Рисунок 8. Процесс создания ЭП в выбранных документах

После того, как закончится процесс создания ЭП, мастер отобразит список обработанных документов и результаты создания ЭП. Для перехода к последней странице мастера нажмите на кнопку **Далее**.

Шаг 8. Завершение работы мастера

На последней странице мастера будет выведена информация об общем количестве документов, количестве успешно подписанных и не подписанных документов (см. *Рисунок 9*).

Для просмотра подробной информации о работе мастера при подписании данного пакета документов, нажмите на кнопку **Просмотреть журнал**.

Для закрытия мастера, нажмите на кнопку **Готово**.

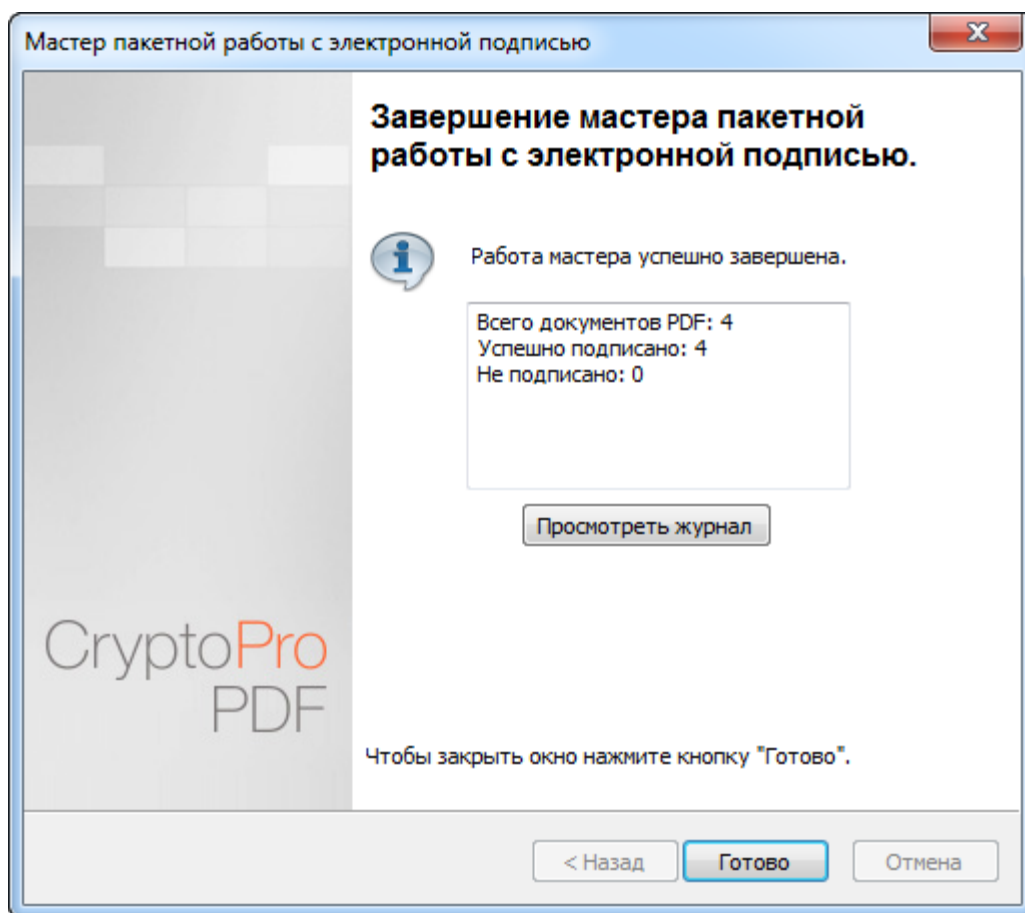


Рисунок 9. Завершение работы мастера

1.1.2. Проверка электронных подписей

Шаг 1. Начало работы мастера

Для проверки ЭП в пакете документов PDF запустите мастер пакетной работы с электронной подписью (**Пуск** ⇒ **Все программы** ⇒ **КРИПТО-ПРО** ⇒ **Пакетная подпись документов PDF**). На экране появится окно мастера (см. *Рисунок 1*). Для пропуска этого шага при следующем запуске мастера, необходимо поставить флаг **Не показывать эту страницу**.

Нажмите на кнопку **Далее**, чтобы продолжить работу мастера.

Шаг 2. Выбор действия с пакетом документов

В этом окне мастера выберите опцию **Проверка электронной подписи в документах** (см. *Рисунок 10*) и нажмите кнопку **Далее**.

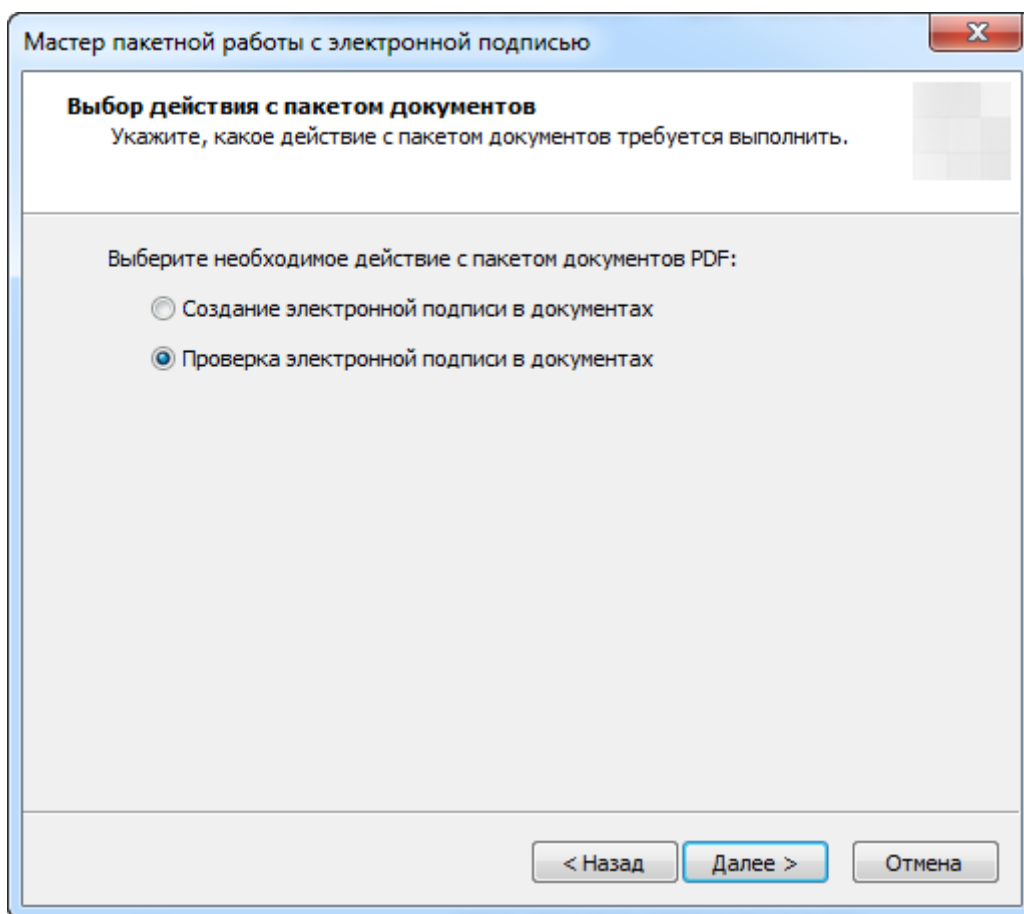


Рисунок 10. Выбор действия с пакетом документов

Шаг 3. Выбор директорий при пакетной проверке ЭП

На данном шаге мастер отобразит диалоговое окно (см. *Рисунок 11*). В поле Директория с документами для проверки подписи необходимо указать директорию с документами для проверки подписи. Это исходная директория, в которой должны содержаться документы PDF, которые необходимо проверить. Все документы в этой директории останутся неизменными.

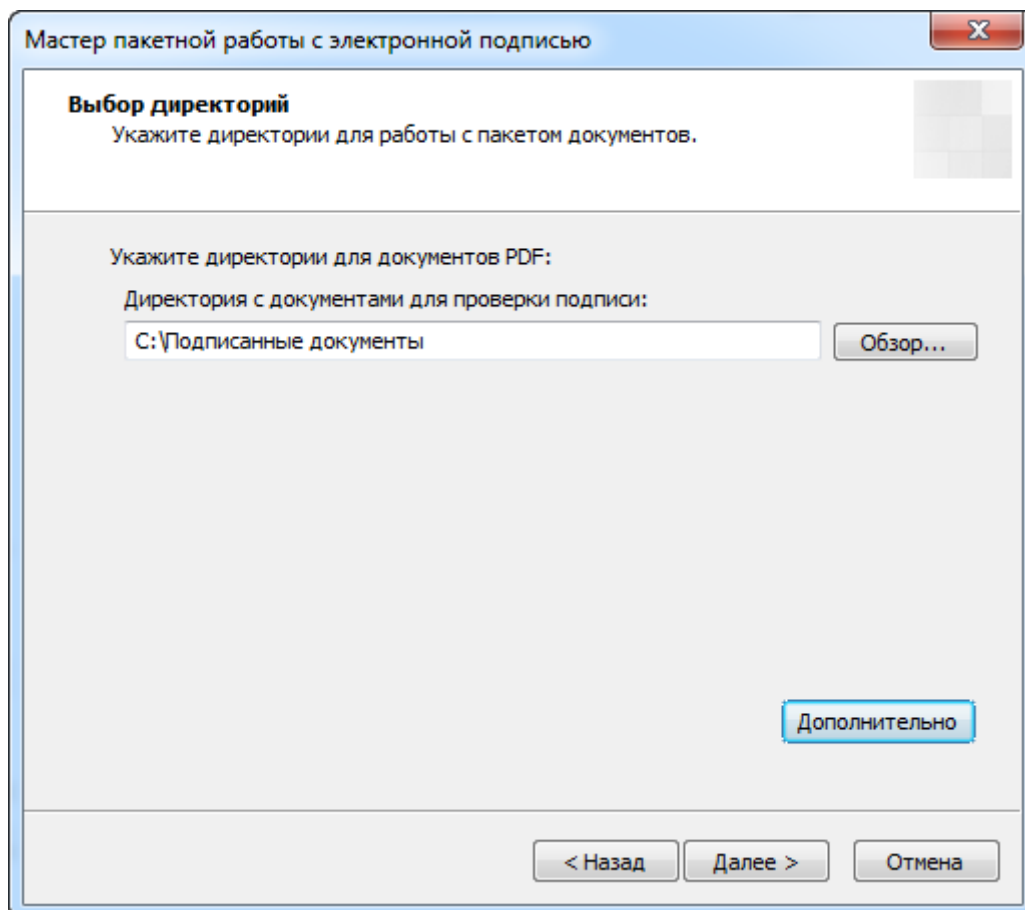


Рисунок 11. Выбор директорий при пакетной проверке ЭП

При нажатии на кнопку **Дополнительно** появится окно с дополнительными настройками (см. *Рисунок 12*).

В появившемся окне можно указать следующие настройки:

- **Директория для журнальных файлов.** В данную директорию будет помещен файл с отчетом о ходе подписи пакета документов PDF.
- **Директория для документов, подпись которых проверить не удалось.** В данную директорию будут помещены копии всех документов PDF из директории с документами для проверки подписи, ЭП в которых проверить не удалось или она была признана недействительной.
- **Директория с проверенными документами.** Это директория, в которую будут помещены копии всех документов PDF из директории с документами для проверки подписи, ЭП в которых были успешно проверены и признаны действительными.
- **Перезаписывать файлы с одинаковыми именами.** При создании файла в директории для проверенных документов или в директории для документов, которые не удалось проверить, уже может существовать файл с таким же именем. Если данная опция включена, то при совпадении имен файл будет перезаписан на новый. Если данная опция отключена, то при совпадении имен процесс создания подписи документе будет прерван.

- **Продолжать работу при возникновении ошибок.** Если данная опция отключена, то при возникновении ошибки программа прекратит процесс создания ЭП и оставшиеся документы PDF подписаны не будут.

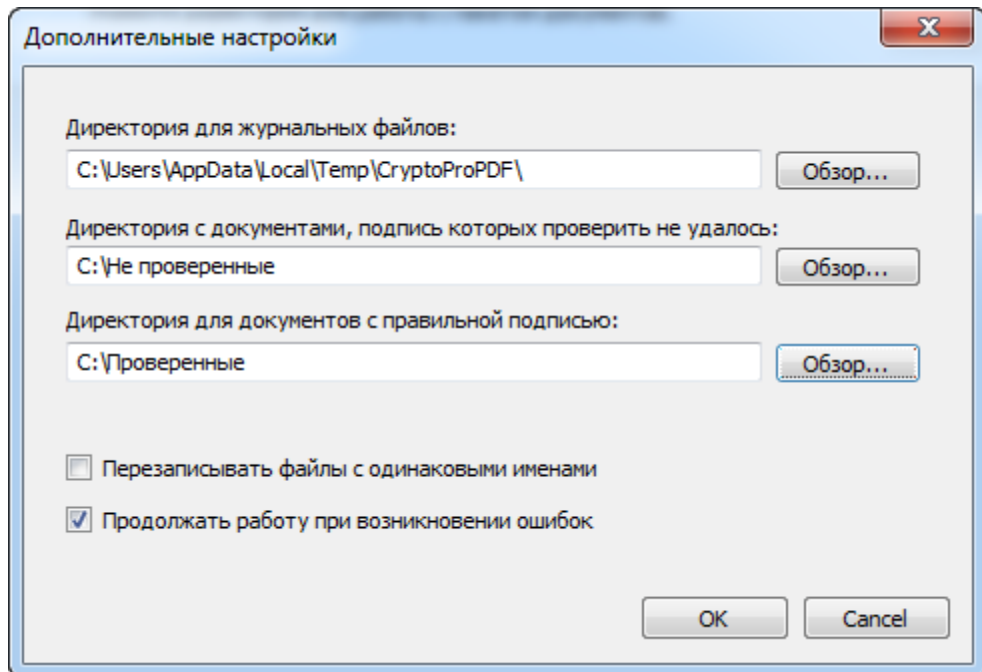


Рисунок 12. Дополнительные настройки

Шаг 4. Выбор параметров подписи

На данном этапе необходимо указать параметры подписи (см. *Рисунок 13*). В каждом из документов PDF будет проверена только одна подпись, параметры которой соответствуют заданным, при этом остальные подписи не проверяются. Если в документе нет подписи с указанными параметрами, то будет возвращена ошибка о том, что подпись не найдена. Если подпись с указанными параметрами существует, то будет возвращен результат проверки.

Поиск ЭП производится по следующим параметрам:

- **Тип подписи** (обычная или сертифицирующая);
- **Соответствие усовершенствованному формату**. Для указания данного параметра необходимо поставить флаг **Проверять соответствие подписи усовершенствованному формату**. В этом случае будет проверена не только действительность ЭП, но и соответствие ее формату CAdES-X Long Type 1, описанному в стандарте «CMS Advanced Electronic Signatures», ETSI Standard, RTS/ESI-000040, TS 101 733».
- **Сертификат, на котором должна быть сформирована ЭП**. Сертификат можно выбрать из хранилища «Личные» текущего пользователя. Для выбора сертификата необходимо нажать на кнопку **Выбрать из хранилища**. Просмотреть выбранный сертификат можно при помощи кнопки **Просмотр**.

Для запуска процесса проверки ЭП в выбранных документах PDF, нажмите на кнопку **Запуск**.

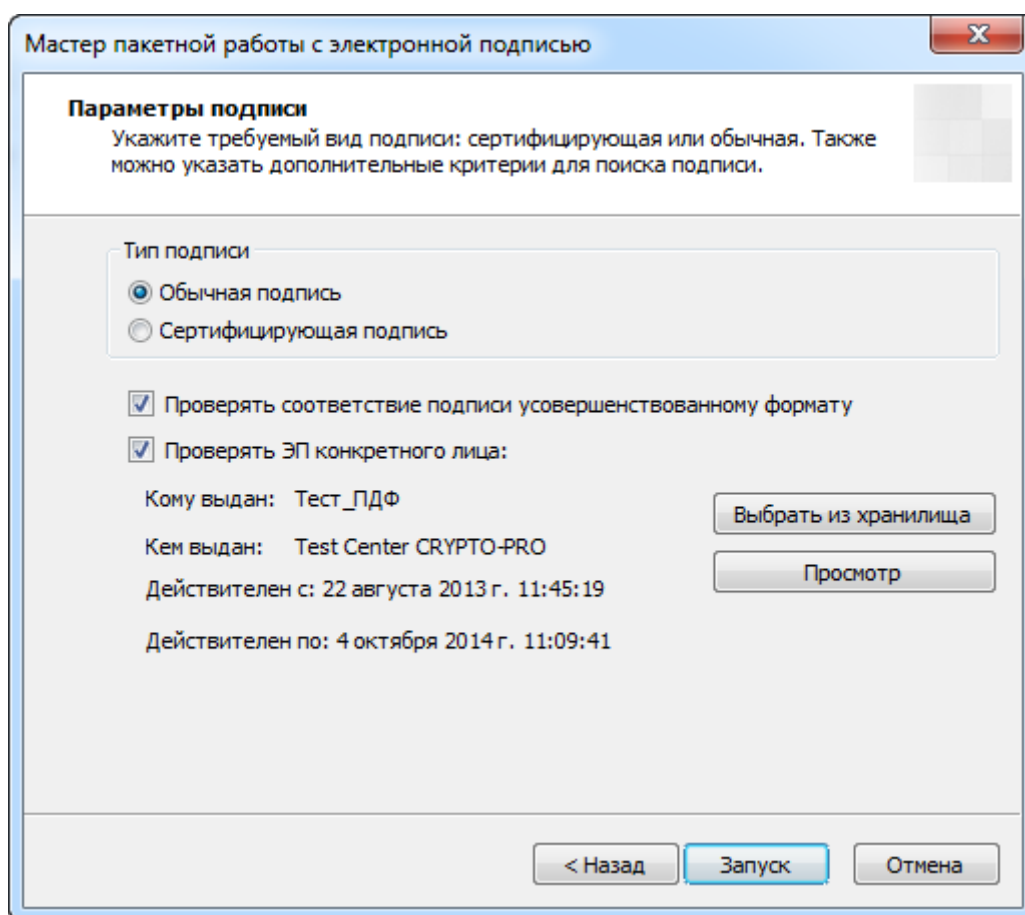


Рисунок 13. Выбор типа подписи и сертификата

Шаг 5. Проверка электронной подписи

На данном шаге мастера запускается процесс проверки подписи в выбранных документах PDF. Если опция **Продолжать работу при возникновении ошибок** (см. *Рисунок 12*) отключена, то при возникновении ошибки программа прекратит процесс проверки ЭП и оставшиеся документы PDF проверены не будут.

После того, как закончится процесс проверки подписей в пакете документов, мастер отобразит перечень обработанных документов и результаты проверки ЭП (см. *Рисунок 14*). Для перехода к последней странице мастера нажмите на кнопку **Далее**.

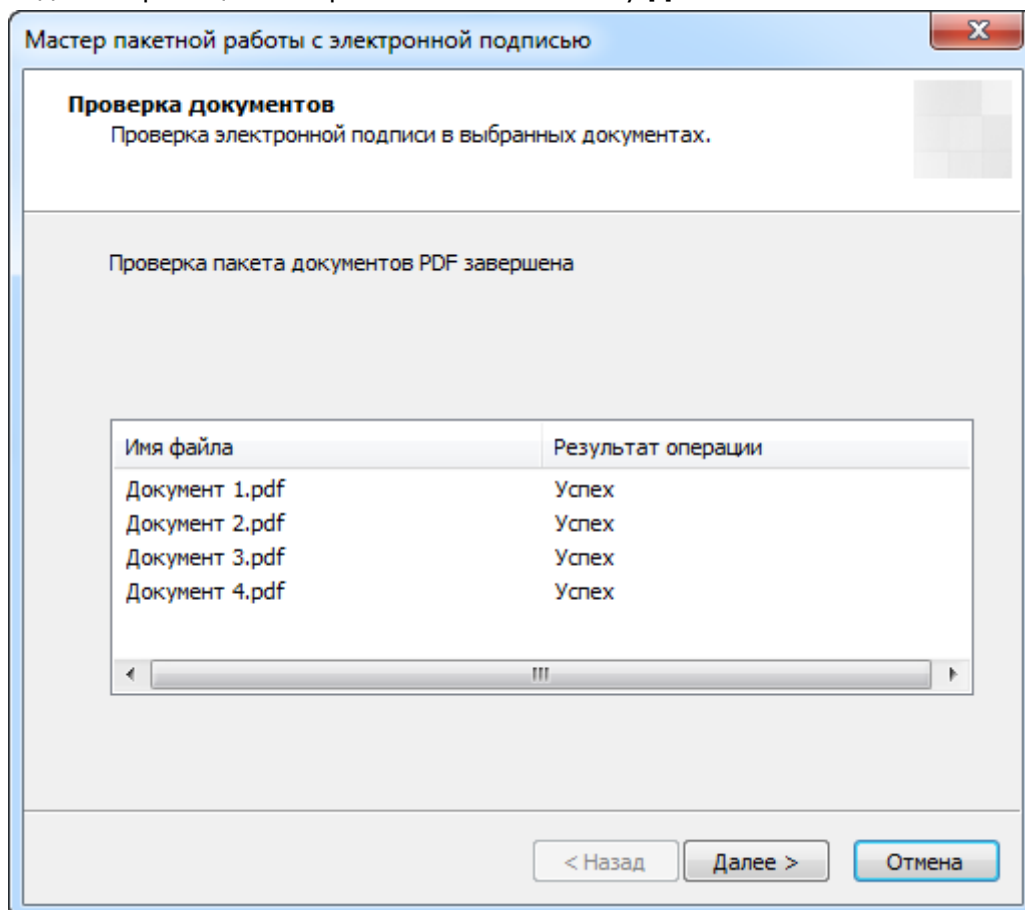


Рисунок 14. Результаты проверки ЭП

Шаг 6. Завершение работы мастера

На последней странице мастера будет выведена информация об общем количестве документов, количестве успешно проверенных и не проверенных документов (см. *Рисунок 15*).

Для просмотра подробной информации о работе мастера при проверке данного пакета документов, нажмите на кнопку **Просмотреть журнал**.

Для закрытия мастера, нажмите на кнопку **Готово**.

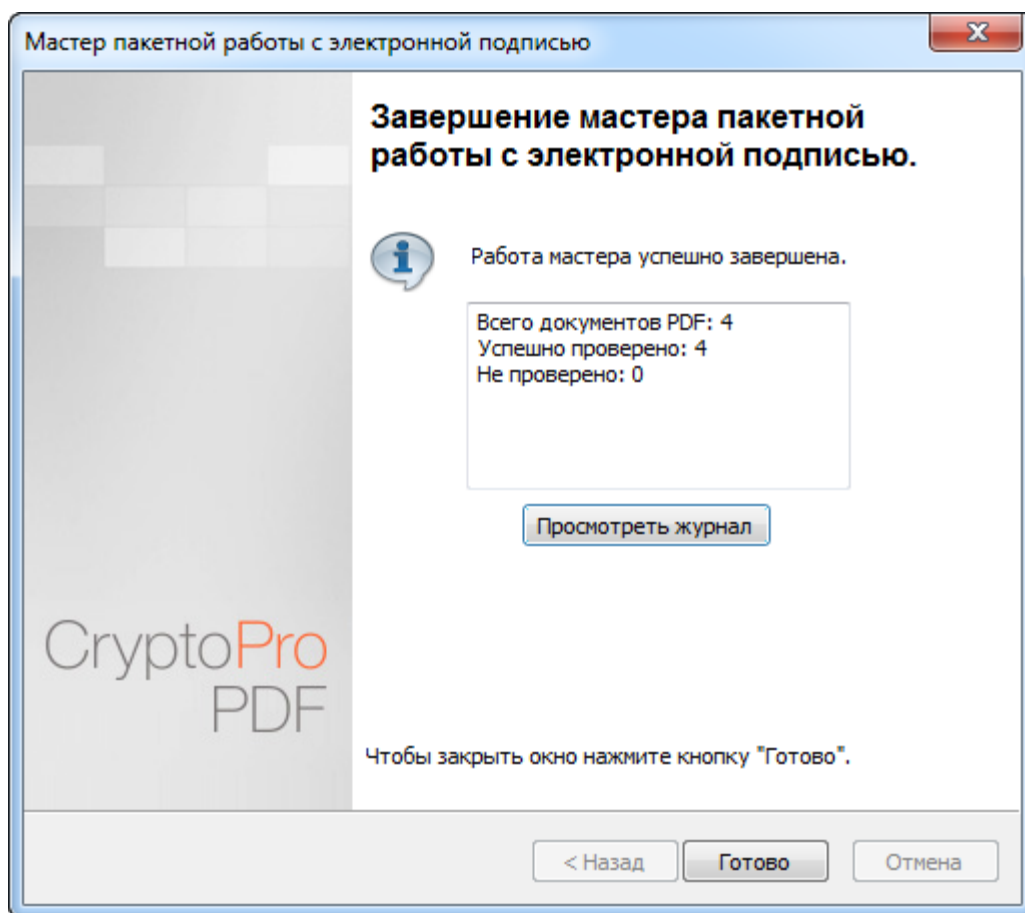


Рисунок 15. Завершение работы мастера при проверке ЭП

1.2. Программа командной строки cprpdfutil.exe

Программа командной строки cprpdfutil.exe предназначена для создания или проверки ЭП в пакете документов PDF. Программа устанавливается в каталог **%ProgramFiles(x86)%\Crypto Pro\PDF**.

1.2.1. Создание электронных подписей

Для запуска процесса создания ЭП в пакете документов PDF, выполните следующую команду:

```
> cprpdfutil.exe sign [параметры]
```

Список параметров команды **sign** приведен в таблице (см. *Таблица 1*).

Таблица 1. Описание параметров команды sign

Параметр	Описание
<i>Обязательные параметры:</i>	
in-dir	Директория с документами, которые необходимо подписать. Все документы в этой директории останутся неизменными.
out-dir	Директории для подписанных документов. В данную директорию будут помещены подписанные копии всех документов PDF из директории с документами на подпись.
certificate	Файл сертификата, который будет использован для создания ЭП. Сертификат должен быть установлен в хранилище «Личные» текущего пользователя и иметь ссылку на закрытый ключ.
<i>Необязательные параметры:</i>	
err-dir	Путь к директории для документов, которые не удалось подписать. В данную директорию будут помещены копии всех документов PDF из директории с документами на подпись, которые не удалось подписать.
report-dir	Директория для журнальных файлов, в которую будет помещен файл отчета о ходе подписи пакета документов PDF. По умолчанию журнальные файлы сохраняются в директорию %TEMP%\CryptoProPDF\.
pin	PIN-код для доступа к закрытому ключу.
ca-des	Если указан данный параметр, то будет создана усовершенствованная ЭП.
url	URL службы штампов времени.
sign-reason	Причина подписи.
location-info	Местоположение.
contact-info	Контактная информация.
certify	Признак создания сертифицирующей подписи.
lock	Признак блокирования документа после создания обычной подписи.
perm	Действия с документом, разрешённые после создания сертифицирующей подписи. Данный параметр может принимать следующие значения: <i>no</i> – изменения запрещены; <i>fillforms</i> – заполнение форм и использование цифровых подписей; <i>comments</i> – комментарии, заполнение форм и использование цифровых подписей. По умолчанию будет использовано значение <i>no</i> .

Параметр	Описание
skip-errors	Если данный параметр не указан, то при возникновении ошибки программа прекратит процесс создания ЭП и оставшиеся документы PDF не будут подписаны.
overwrite-files	Если указан данный параметр, то при совпадении имени создаваемого файла с одним из имен в директориях out-dir или err-dir, существующий файл будет перезаписан на новый.
help	Вызов справки по команде sign.

1.2.2. Проверка электронных подписей

Для запуска процесса проверки ЭП в пакете документов PDF, выполните следующую команду:

```
> cspdfutil.exe verify [параметры]
```

Список параметров команды **verify** приведен в таблице (см. Таблица 2).

Таблица 2. Описание параметров команды verify

Параметр	Описание
<i>Обязательные параметры:</i>	
in-dir	Директория с документами для проверки подписи. Все документы в этой директории останутся неизменными.
<i>Необязательные параметры:</i>	
out-dir	Директории с проверенными документами. В данную директорию будут помещены копии документов PDF из директории с документами для проверки подписи, в которых была найдена успешно проверенная и действительная ЭП, параметры которой соответствуют заданным.
err-dir	Директория для документов, подпись которых проверить не удалось. В данную директорию будут помещены копии всех документов PDF из директории с документами для проверки подписи, в которых ЭП в которых проверить не удалось или она была признана недействительной.
report-dir	Директория для журнальных файлов, в которую будет помещен файл отчета о ходе проверки пакета документов PDF. По умолчанию журнальные файлы сохраняются в директорию %TEMP%\CryptoProPDF\.
req-certificate	Файл сертификата, наличие и действительность ЭП которого следует проверить в документах PDF. Если указан данный параметр, то возвращается статус подписи, сделанной на данном сертификате, при этом остальные подписи не проверяются.
req-cades	Если указан данный параметр, то будет проверена только подпись, соответствующая усовершенствованному формату (CAES-X Long Type 1, описанному в стандарте "CMS Advanced Electronic Signatures", ETSI Standard, RTS/ESI-000040, TS 101 733).
req-certified	Если указан данный параметр, то в каждом документе PDF будет проверена только сертифицирующая подпись, при этом обычные подписи не проверяются.
skip-errors	Если данный параметр не указан, то при возникновении ошибки программа прекратит процесс проверки ЭП и оставшиеся документы PDF проверены не будут.

Параметр	Описание
overwrite-files	Если указан данный параметр, то при совпадении имени создаваемого файла с одним из имен в директориях out-dir или err-dir, существующий файл будет перезаписан на новый.
help	Вызов справки по команде verify.

Примечание. В каждом из документов PDF будет проверена только одна подпись, параметры которой соответствуют заданным, при этом остальные подписи не проверяются. Например, если одновременно указаны параметры *req-cades*, *req-certified*, *req-certificate*, то в документе ищется усовершенствованная сертифицирующая ЭП, сделанная на данном сертификате, и проверяется ее действительность.

2. Автоматизация создания и проверки подписи

2.1. Автоматизация с использованием Adobe JavaScript

В программе Adobe Acrobat имеется возможность использовать сценарии JavaScript для формирования и проверки ЭП. С помощью сценариев можно обеспечить автономную работу с ЭП без привлечения графического пользовательского интерфейса.

Для включения возможности использования сценариев JavaScript, необходимо выполнить следующие действия:

1. Откройте главное окно программы Adobe Acrobat.
2. Выберите в разделе **Редактирование (Edit)** пункт **Установки (Preferences)**.
3. В открывшемся окне (см. *Рисунок 16*) выберите вкладку **JavaScript** и нажмите флаг **Активировать Acrobat JavaScript (Enable Acrobat JavaScript)**.

По умолчанию данная опция активирована при установке Adobe Acrobat.

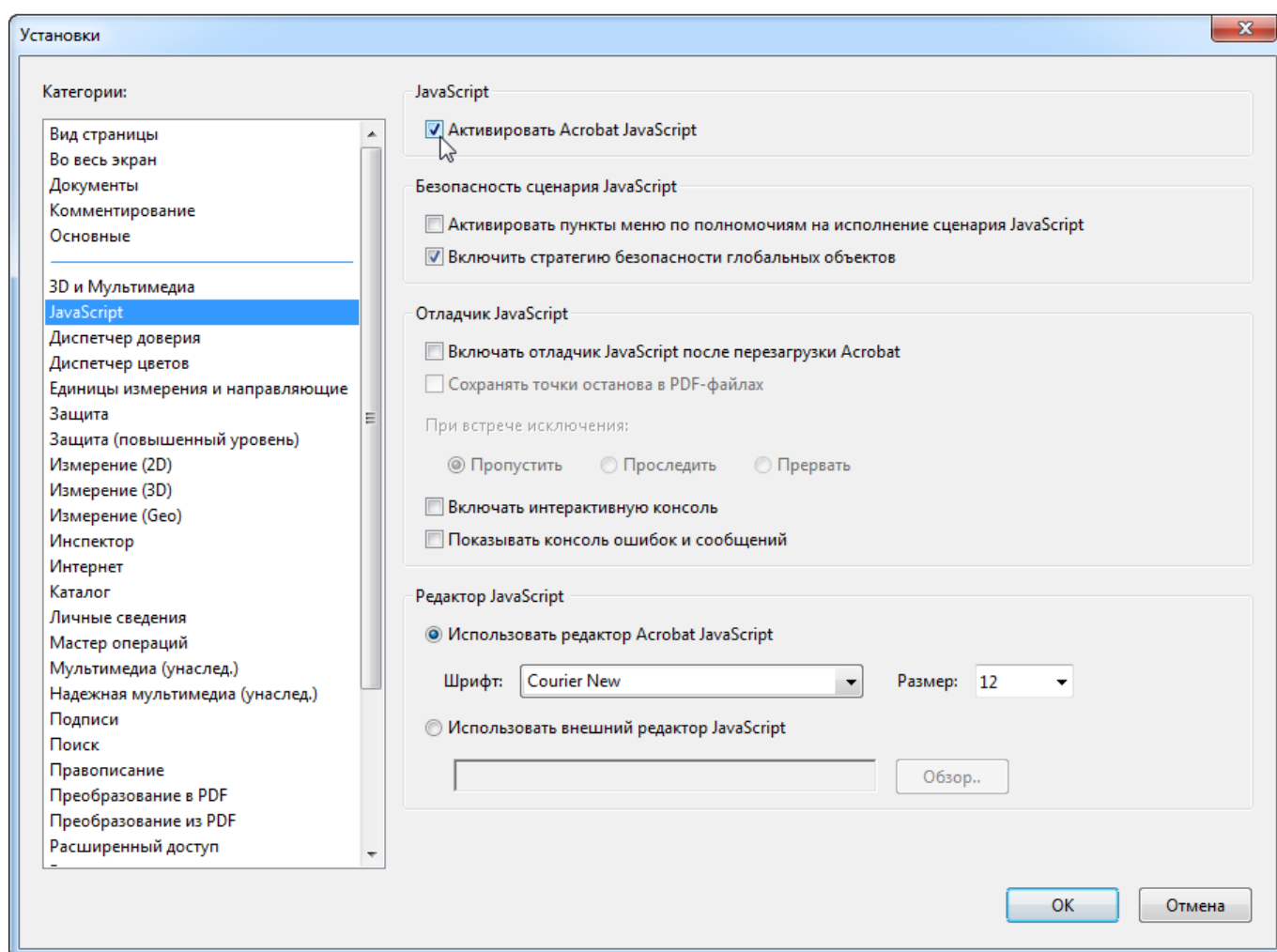


Рисунок 16. Настройка JavaScript в Adobe Acrobat

При запуске программы Adobe Acrobat будут автоматически загружены все доступные файлы с расширением *.js из директории **<путь установки приложения Adobe Acrobat> \Acrobat\Javascrpts**. Код JavaScript, который содержится в данных файлах, будет доступен для исполнения в программе Adobe Acrobat.

2.1.1. Пример кода для формирования электронной подписи

Ниже приведен пример кода JavaScript, осуществляющий формирование невидимой ЭП в документе:

```
// объект текущего документ
var doc = this;
// создать объект невидимой подписи
var f = doc.addField('MySignature', 'signature', 0, [0,0,0,0]);
// получить обработчик КриптоПро PDF
var engine = security.getHandler('CryptoPro PDF');
// загрузить сертификат из файла c:\path\mycert.cer
var c = security.importFromFile({ cType:'Certificate',
  cDIPath:'/c/path/mycert.cer', bUI:false });
// инициализация
engine.login({ oParams:{ oEndUserSignCert:c } });
// дополнительная информация
var s = { reason: 'MyReason', location: 'Moscow' };
// вычисление ЭП
f.signatureSign({ oSig: engine, oInfo: s, bUI: false })
```

2.1.2. Пример кода для проверки электронной подписи

Ниже приведен пример кода JavaScript, осуществляющий проверку и вывод информации об ЭП в документе:

```
var doc = this;
// получить объект существующей подписи
var f = doc.getField('MySignature');
// проверить ЭП
var res = f.signatureValidate();
// анализ результата проверки
switch(res)
{
  case -1: console.println('Not a signature field'); break;
  case 0: console.println('Signature is blank'); break;
  case 1: console.println('Unknown status'); break;
  case 2: console.println('Signature is invalid'); break;
  case 3: console.println('Signature of document is valid, identity of signer
could not be verified'); break;
  case 4: console.println('Signature of document is valid and identity of
signer is valid'); break;
  default: console.println('Undefined result');
}
// вывод информации о подписи
var sinfo = f.signatureInfo();
console.println('Signer: ' + sinfo.name + ' ' + sinfo.contactInfo);
```


2.1.3. Отладка и выполнение кода JavaScript в программе Adobe Acrobat

Для отладки и выполнения сценария JavaScript можно воспользоваться отладчиком JavaScript.

Для запуска отладчика JavaScript, необходимо выполнить следующие действия:

1. Откройте главное окно программы Adobe Acrobat.
2. Выберите в разделе **Просмотр (View)** пункт **Инструменты (Tools)**, затем **JavaScript** (см. *Рисунок 17*).
3. В главном окне Adobe Acrobat появится вкладка, на которой необходимо нажать **Отладчик JavaScript (JavaScript Debugger)**.

Для открытия отладчик JavaScript с помощью клавиатуры, нажмите комбинацию клавиш **CTRL+J**.

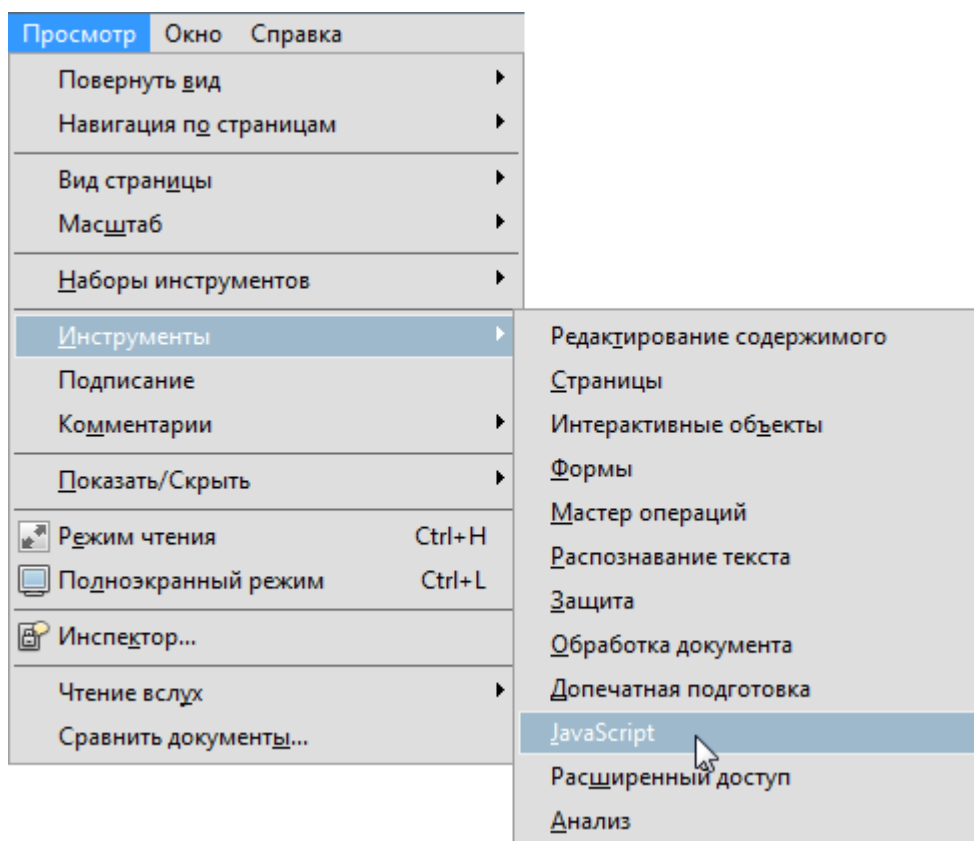


Рисунок 17. Запуск отладчика JavaScript

Для выполнения строки кода в JavaScript Debugger, необходимо поместить курсор в эту строку и нажать **CTRL+ENTER** (см. *Рисунок 18*).

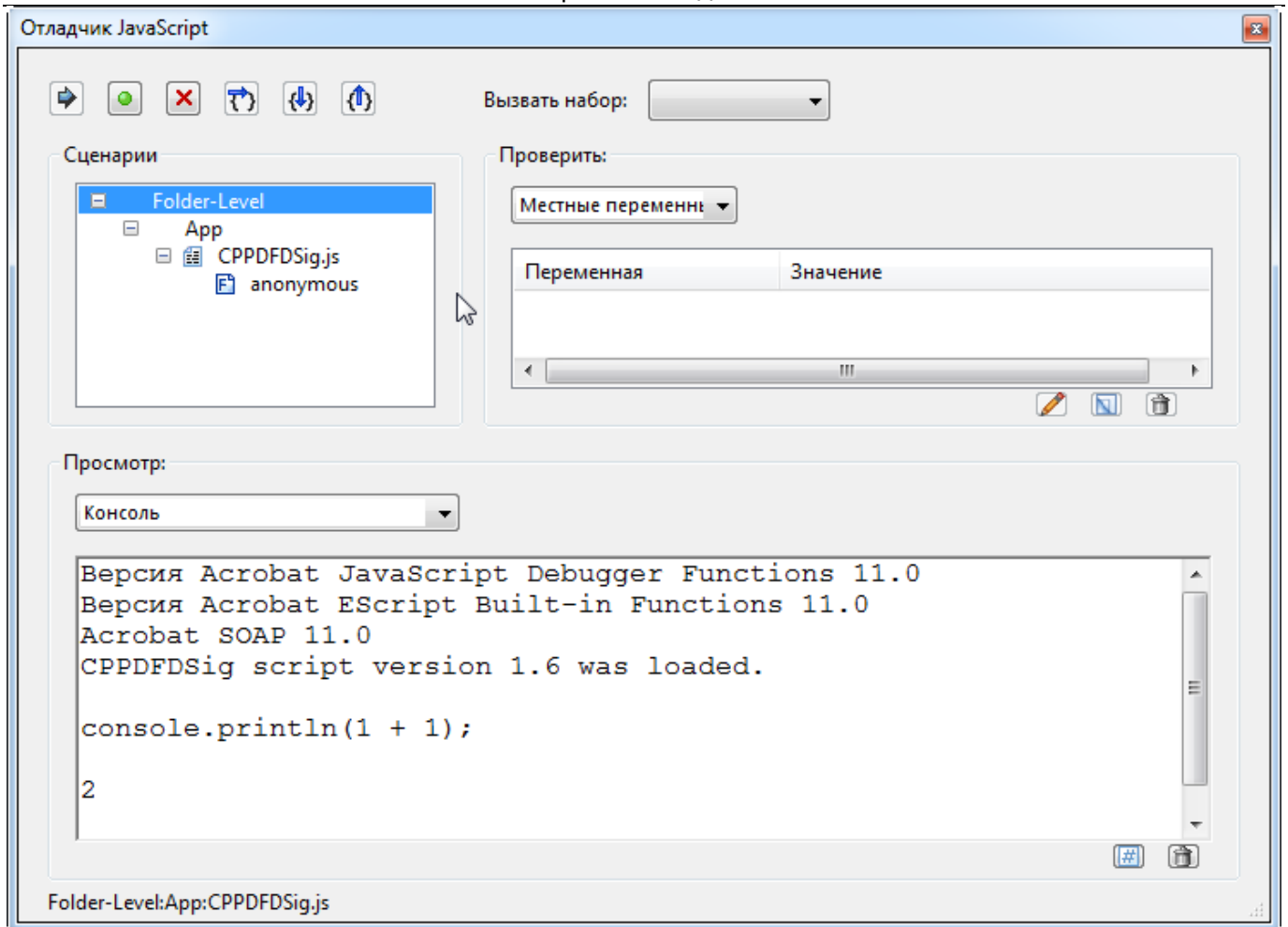


Рисунок 18. Отладчик JavaScript

Подробное описание использования JavaScript в Adobe Acrobat можно найти на сайте Adobe в разделе для разработчиков <http://www.adobe.com/devnet/acrobat/?view=documentation>.

Таблица 3. Краткий обзор функций JavaScript для работы с КриптоПро PDF

Название метода	Описание
security.getHandler	<p>Функция получает объект для взаимодействия с криптографическим модулем.</p> <p>Параметры: <i>cName</i> – Имя криптографического модуля. Для модуля «КриптоПро PDF» значение должно быть «CryptoPro PDF».</p> <p>Возвращаемое значение: Возвращает объект SecurityHandler.</p>
SecurityHandler.login	<p>Функция идентификации пользователя.</p> <p>Параметры: <i>oParams</i> – Объект LoginParameters, который содержит параметры для идентификации пользователя. <i>bUI</i> – Устанавливается <i>true</i>, если в процессе идентификации пользователя должен использоваться графический интерфейс. В противном случае устанавливается значение <i>false</i>.</p> <p>Возвращаемое значение: Возвращает <i>true</i> при успешной идентификации, иначе <i>false</i>.</p>
Field.signatureSign	<p>Функция формирования ЭП в созданном поле документа.</p> <p>Параметры: <i>oSig</i> – Объект SecurityHandler связанный с криптографическим модулем. Если данный параметр отсутствует, то будет выбран модуль, указанный в настройках Adobe Acrobat или будет предложен выбор модуля через графический интерфейс (если параметр <i>bUI</i> имеет значение <i>true</i>). <i>oInfo</i> – Объект SignatureInfo с информацией об ЭП. <i>bUI</i> – Флаг, определяющий нужно ли показывать графический интерфейс во время формирования ЭП. Если установлено значение <i>true</i>, то для заполнения полей диалогов формирования подписи используется объект <i>oInfo</i>. Если установлено значение <i>false</i>, процесс подписания происходит без графического интерфейса.</p> <p>Возвращаемое значение: Возвращает <i>true</i>, если подпись была создана удачно. В противном случае возвращает <i>false</i>.</p>

Таблица 4. Краткий обзор объектов JavaScript для работы с КриптоПро PDF

Название объекта	Описание
LoginParameters	<p>Параметры для идентификации пользователя:</p> <p><i>oEndUserSignCert</i> – Объект Certificate, содержащий сертификат пользователя для формирования ЭП.</p> <p><i>cPassword</i> – PIN код для доступа к контейнеру закрытого ключа.</p>
SignatureInfo	<p>Свойства подписи:</p> <p><i>appearance</i> – Вид подписи. Указывается одно имя из тех, которые установлены в настройках Adobe Acrobat в разделе Защита.</p> <p><i>certificates</i> – Массив, содержащий иерархию сертификатов (объектов Certificate), которые идентифицируют пользователя. Первый сертификат в этом множестве – это сертификат пользователя, последующие являются сертификатами удостоверяющих центров.</p> <p><i>contactInfo</i> – Дополнительная контактная информация о пользователе (телефон, адрес и т.п.)</p> <p><i>date</i> – Объект <i>Date</i>. Дата подписания.</p> <p><i>dateTrusted</i> – Значение, показывающее получена ли дата из доверенного источника (имеется штамп времени). Значение <i>false</i> – если дата не является доверенной и взята с локального компьютера пользователя.</p> <p><i>handlerName</i> – Имя криптографического метода.</p> <p><i>mdp</i> (Modification Detection and Prevention) - Задаёт допустимые изменения в документе после его подписания. Может принимать следующие значения:</p> <ul style="list-style-type: none"> <i>allowNone</i> – запрещаются какие-либо изменения; <i>allowAll</i> – допускается редактирование документа; <i>default</i> – разрешается редактировать поля формы и создавать ЭП; <i>defaultAndComments</i> – разрешается редактировать поля формы, создавать ЭП и добавлять комментарии. <p><i>name</i> – Имя подписавшего пользователя.</p> <p><i>reason</i> – Причина подписания.</p> <p><i>status</i> – Статус проверки ЭП, выполненной функцией signatureValidate.</p> <p><i>statusText</i> – Сообщение с результатом проверки ЭП, выполненной функцией signatureValidate.</p> <p><i>timeStamp</i> – URL сервера штампов времени. Если данный параметр задан, то создается усовершенствованная ЭП в соответствии со стандартом CAdES («CMS Advanced Electronic Signatures», ETSI Standard, RTS/ESI-000040, TS 101 733).</p>

2.2. Автоматизация с использованием iText

iText — это библиотека с открытым исходным кодом для генерации, анализа и изменения документов в формате PDF. Изначально библиотека iText была разработана для языка Java, а затем портирована под платформу Microsoft .NET Framework в виде библиотеки iTextSharp, написанной на C#.

Позже появилась библиотека iTextSharp, написанная на C# и представляющая собой порт iText под платформу Microsoft .NET Framework.

Более подробную информацию о данных продуктах можно найти на сайте компании iText Software Corp.:

<http://itextpdf.com/product>

2.2.1. Разработка на языке Java

Для автоматизации создания и проверки ЭП на языке Java необходимо использовать библиотеку iText совместно с КриптоПро JCP.

Подробную информацию и примеры программного кода можно найти на форуме КРИПТО-ПРО по адресу:

<http://www.cryptopro.ru/forum2/default.aspx?q=posts&t=4205#post23002>

2.2.2. Разработка на платформе Microsoft .NET Framework

Для автоматизации создания и проверки ЭП на платформе Microsoft .NET Framework необходимо использовать библиотеку iTextSharp совместно с продуктами КриптоПро CSP и КриптоПро .NET.

Тексты примеров программного кода для формирования и проверки электронной подписи представлены в составе КриптоПро .NET SDK (файл Simple35.zip, который по умолчанию располагается в каталоге %ProgramFiles(x86)%\Crypto Pro\.NET SDK\Examples).

Подробную информацию можно найти в документации КРИПТО-ПРО следующим адресам:

<http://cpdn.cryptopro.ru/default.asp?url=content/cpnet/html/a6189c38-f349-40f0-bdf0-0a620d91efbb.htm>

<http://cpdn.cryptopro.ru/default.asp?url=content/cpnet/html/d5da1068-67b6-4247-bb41-4d1bd1e77774.htm>

2.3. Автоматизация с использованием JavaScript и OLE

Плагин «КриптоПро PDF» содержит сценарий JavaScript для автоматизации создания и проверки подписи в документах PDF. Данный сценарий содержится в файле CPPDFDSig.js, который после установки плагина находится в директории **<путь установки приложения Adobe Acrobat>\Acrobat\Javascrpts**.

В сценарии представлены два метода: для создания подписи *CPSign* и для проверки подписи *CPVerify*.

```
CPSign(  
    certificateHEX,  
    pin,  
    pdfLockMode,  
    signReason,  
    signLocation,  
    signContactInfo,  
    timeStamp,  
    bShowCertificatesDialog,  
    bShowSignDialogs,  
    appearance  
);
```

Описание

Функция подписывает PDF документ.

Параметры

Таблица 5. Параметры функции *CPSign*

Параметр	Тип	Описание
certificateHEX	String	Обязательный параметр. Массив байт сертификата, используемого для создания ЭП, в виде строки шестнадцатеричных цифр. Сертификат должен быть установлен в хранилище «Личные» текущего пользователя и иметь ссылку на закрытый ключ.
pin	String	PIN-код для доступа к закрытому ключу.
pdfLockMode	Numeric	Число, характеризующее тип создаваемой подписи и действия, разрешенные с документом после подписания: 0 — обычная подпись, изменения разрешены; 1 — обычная подпись, документ заблокирован; 2 — сертифицирующая подпись, изменения запрещены; 3 — сертифицирующая подпись, разрешается заполнение полей форм и создание ЭП; 4 — сертифицирующая подпись, разрешается добавление комментариев, заполнение полей форм и создание ЭП.
signReason	String	Причина подписи.

signLocation	String	Местоположение.
signContactInfo	String	Контактная информация.
timeStamp	String	URL службы штампов времени. Если данный параметр задан, то будет сделана усовершенствованная подпись.
bShowCertificatesDialog	Boolean	Флаг, указывающий будут ли отображены диалог со списком сертификатов и диалог для ввода ПИН-кода.
bShowSignDialogs	Boolean	Флаг, указывающий будут ли отображены диалог с параметрами подписи и диалог для сохранения файла.
appearance	Object	Объект, определяющий параметры поля подписи и его содержимое (см. раздел 3). Если параметр не задан, то будет создана невидимая подпись.

Возвращаемое значение

При успешном подписании документа возвращаемое значение равно `true`. Иначе возвращается строка с описанием ошибки.

```
CPVerify(
    reqCAAdES,
    reqCertified,
    certificateHEX,
    bShowVerifyDialogs
);
```

Описание

Функция проверяет подпись в PDF документе.

Параметры

Таблица 6. Параметры функции `CPVerify`

Параметр	Тип	Описание
reqCAAdES	Boolean	Флаг, указывающий должна ли проверяемая подпись соответствовать усовершенствованному формату.
reqCertified	Boolean	Флаг, указывающий должна ли проверяемая подпись быть сертифицирующей.
certificateHEX	String	Массив байт сертификата, наличие и действительность ЭП которого следует проверить в документах PDF, в виде строки шестнадцатеричных цифр.
bShowVerifyDialogs	Boolean	Флаг, указывающий будет ли отображен диалог со списком сертификатов.

Возвращаемое значение

Если подпись с заданными параметрами существует и верна, то возвращаемое значение равно `true`. В остальных случаях возвращается строка с описанием статуса подписи или ошибки.

Данные методы можно использовать в своих сценариях JavaScript, а также при взаимодействии сторонних приложений и Adobe Acrobat с помощью технологии OLE. Более подробную информацию об этом можно найти в документации Adobe:

http://livedocs.adobe.com/acrobat_sdk/11/Acrobat11_HTMLHelp/IAC_DevApp_OLE_Support.101.13.html

Примеры на выполнение сценариев JavaScript из приложений, написанных на языках Visual Basic, C++, C#, можно найти в Acrobat SDK в папке InterAppCommunicationSupport.

3. Настройка оформления видимой подписи

С помощью сценария JavaScript, содержащегося в файле CPPDFDSig.js, можно задать расположение видимой подписи, указать предустановленный шаблон оформления или создать настраиваемое оформление видимой подписи. Для этого необходимо проинициализировать объект `appearance` и передать его в качестве параметра в функцию `CPSign`.

Таблица 7. Описание объекта ***appearance***

Свойство	Тип	Описание
<code>pageNumber</code>	Integer	Номер страницы для создания поля подписи начиная с 0. Если параметр не задан, подпись будет отображаться на первой странице.
<code>coords</code>	Array	Массив из четырех чисел, определяющих координаты поля подписи. Координаты должны быть указаны в следующем порядке: крайняя левая точка по оси x, крайняя верхняя точка по оси y, крайняя правая точка по оси x, крайняя нижняя точка по оси y. Координаты поля подписи задаются относительно левого нижнего угла страницы. Если координаты не заданы, то будет создана невидимая подпись.
<code>appearanceFilter</code>	String	Название оформления подписи с фиксированными полями, созданного через Acrobat (см. раздел 3.1). При использовании данного свойства, свойство <code>customAppearance</code> должно остаться неинициализированным.
<code>customAppearance</code>	Object	Объект, содержащий настройки оформления подписи, позволяющий настраивать оформление поля подписи: количество текстовых полей, их содержимое, визуальные характеристики поля подписи (см. раздел 3.2). При использовании данного свойства, свойство <code>appearanceFilter</code> должно остаться неинициализированным.

3.1. Использование предустановленного оформления подписи

Для использования предустановленного оформления подписи, необходимо:

1. Создать оформление подписи с нужными элементами.

Для этого в меню Adobe Acrobat выбрать пункт **Редактирование (Edit)** -> **Установки (Preferences)**. Далее выбрать вкладку **Подписи (Signatures)** -> **Создание и оформление (Creation & Appearance)** -> **Установки создания и оформления (Creation & Appearance Preferences)**, нажать кнопку **"Создать..." (Create)**.

2. Сохранить созданное оформление, указав заголовок (см. Рисунок 19).

3. Проинициализировать свойства объекта `appearance: pageNumber, coords, appearanceFilter`.

4. Передать объект `appearance` в функцию `CPSign()` в качестве параметра.

Ниже приведен пример инициализации объекта `appearance`:

```
var myAppearance =  
{
```

```
pageNumber: 0,  
coords: [122, 539, 422, 439],  
appearanceFilter: "Fancy" // заголовок  
};
```

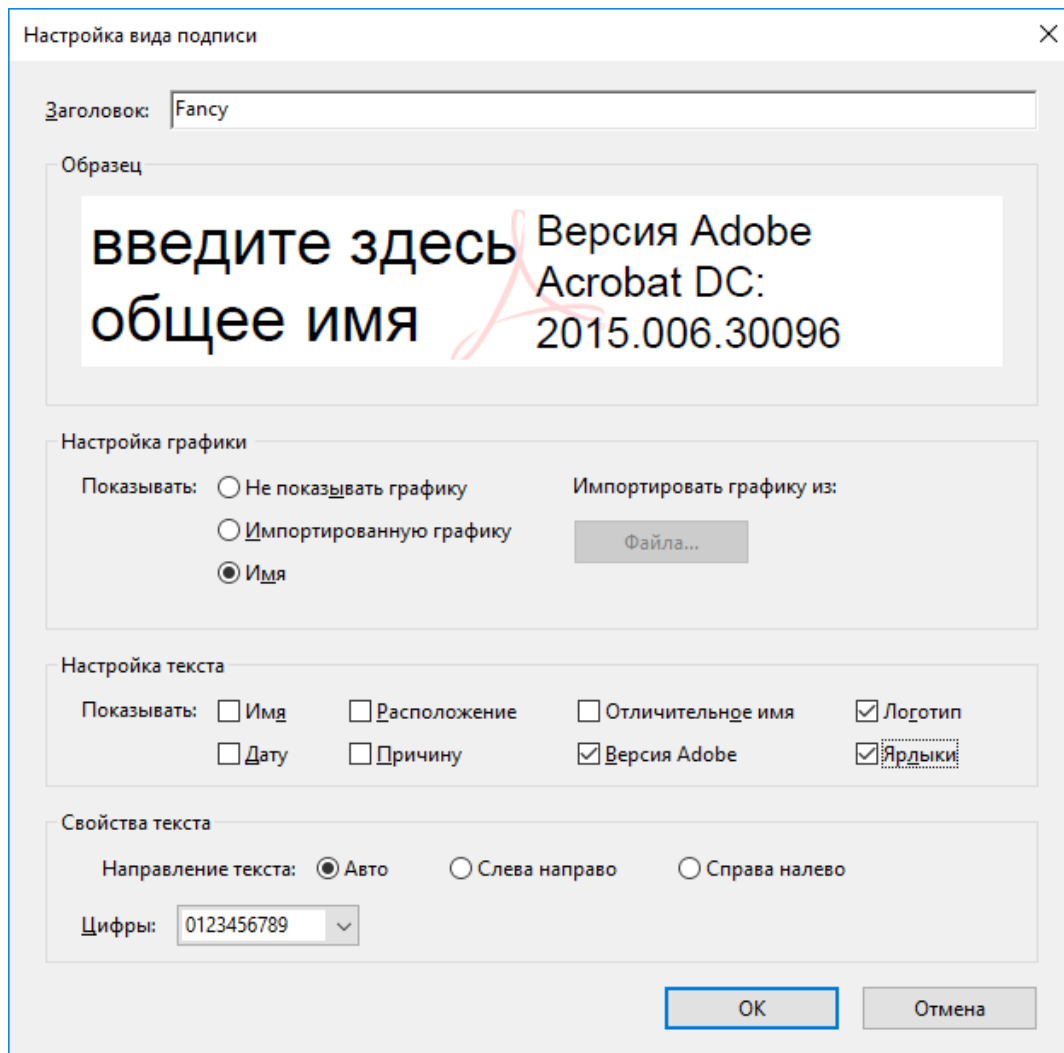


Рисунок 19. Настройка оформления подписи

3.2. Создание настраиваемого оформления подписи

С помощью языка сценариев JavaScript можно создать настраиваемое оформление видимой подписи с необходимыми визуальными характеристиками, количеством текстовых полей, их содержимым. Визуальные характеристики оформления представляют собой шрифт, тип, цвет и толщину границы поля подписи. Текстовые поля могут содержать текст, полученный из сертификата подписи/параметров подписи, или произвольный текст, заданный пользователем.

Для настройки оформления подписи необходимо:

1. Проинициализировать объект `appearance` свойствами `pageNumber`, `coords`, `customAppearance`.
2. Вызвать функцию `CPSign` с данным параметром.

Таблица 8. Описание объекта **customAppearance**

Свойство	Тип	Описание
font	String	Строка, содержащая название шрифта. Если шрифт с указанным названием не существует, то будет использовано значение по умолчанию.
borderStyle	Keyword	Определяет тип границы вокруг поля подписи. Возможные значения: border.s - сплошная граница; border.b - сплошная граница со скошенными краями, border.d - пунктирная граница, border.i - сплошная граница со скошенными внутрь краями; border.u - граница с подчеркиванием по нижнему краю поля подписи.
borderColor	Array / keyword	Массив, содержащий 1, 2, 4 или 5 элементов. Первый элемент является строкой, задающей цветовое пространство ("T", "G", "RGB", "CMYK"), следующие элементы задают компоненты цвета. Также можно использовать ключевое слово, описывающее цвет. Например, ключевое слово для красного цвета - color.red, соответствующий массив - ["RGB", 1, 0, 0]. Более подробно см. JavaScript for Acrobat API Reference.
borderWidth	Integer	Ширина границы. Возможные значения: 0 – границы нет; 1 – тонкая; 2 – средняя; 3 – толстая.
textFields	Array	Массив объектов textField, содержащих параметры отображения текстовых полей.

Таблица 9. Описание объекта **textField**

Свойство	Тип	Описание
text	String	Строка, содержащая текст для отображения в данном поле. Может содержать символы табуляции. Служит для задания тестовых полей с произвольным содержанием, заголовков. При использовании данного свойства, свойство type должно остаться непроинициализированным.

type	String	Определяет содержимое тестового поля, которое берется из сертификата подписи или дополнительных из параметров подписи. При использовании данного свойства, свойство <code>text</code> должно остаться непроинициализированным. Возможные значения: "CN" – владелец сертификата; "DN" – субъект; "ValidityPeriod" – срок действия сертификата; "SerialNumber" – серийный номер сертификата; "Reason" – причина подписи; "Location" – местоположение.
coords	Object	Объект, содержащий координаты текстового поля. Значения координат задаются относительно левого нижнего угла поля подписи.
fontSize	Integer	Определяет размер шрифта. Может принимать значения от 0 до 32767 включительно. Если значение не указано или равно 0, то будет использован максимальный размер шрифта, при котором текст полностью поместится в данное поле.

Ниже приведен пример инициализации объекта `myAppearance`, который необходимо передать в качестве параметра в функцию `CPSign()`:

```
var myAppearance =
{
  pageNumber: 0,
  coords: [122, 539, 422, 439],
  customAppearance:
  {
    font: "Calibri",
    borderStyle: border.s,
    borderColor: color.black,
    borderWidth: 1,
    textFields:
    [
      {
        text: " ДОКУМЕНТ ПОДПИСАН \nЭЛЕКТРОННОЙ ПОДПИСЬЮ",
        coords: [50, 100, 250, 60]
      },
      {
        type: "CN",
        coords: [10, 60, 300, 40],
        fontSize: 11
      }
    ]
  }
}
```

```
    },  
    {  
        type: "SerialNumber",  
        coords: [10, 40, 300, 20],  
        fontSize: 11  
    },  
    {  
        text: "Компания: ООО \"КРИПТО-ПРО\"",  
        coords: [10, 20, 300, 0],  
        fontSize: 11  
    }  
]  
}  
};
```

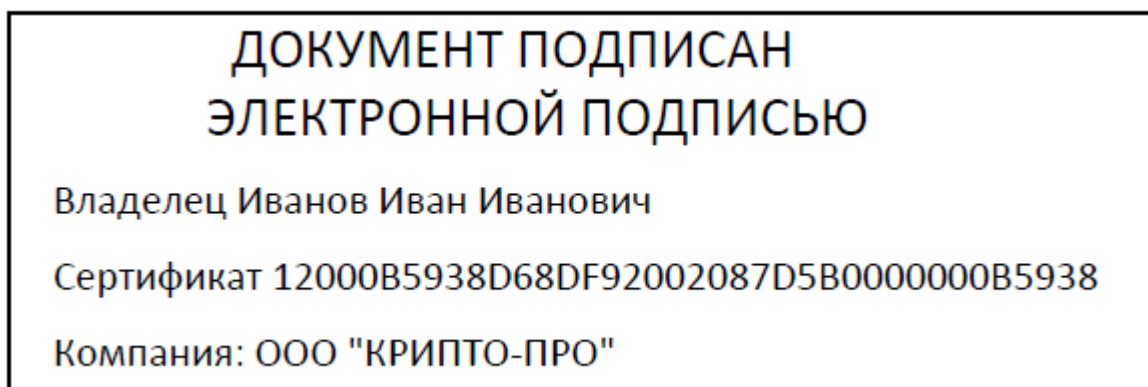


Рисунок 20. Пример настраиваемого оформления подписи

4. Перечень сокращений

СКЗИ	Средство криптографической защиты информации
ЭП	Электронная подпись
IAC	Interapplication Communication
PDF	Portable Document Format, платформонезависимый формат электронных документов
TSP	Time-Stamp Protocol

5. Перечень рисунков

<i>Рисунок 1. Мастер пакетной работы с электронной подписью</i>	<i>5</i>
<i>Рисунок 2. Выбор действия с пакетом документов</i>	<i>6</i>
<i>Рисунок 3. Выбор директорий при пакетном создании ЭП</i>	<i>7</i>
<i>Рисунок 4. Дополнительные настройки</i>	<i>8</i>
<i>Рисунок 5. Выбор сертификата подписи</i>	<i>9</i>
<i>Рисунок 6. Выбор типа подписи</i>	<i>10</i>
<i>Рисунок 7. Параметры подписи</i>	<i>11</i>
<i>Рисунок 8. Процесс создания ЭП в выбранных документах</i>	<i>12</i>
<i>Рисунок 9. Завершение работы мастера</i>	<i>13</i>
<i>Рисунок 10. Выбор действия с пакетом документов</i>	<i>14</i>
<i>Рисунок 11. Выбор директорий при пакетной проверке ЭП</i>	<i>15</i>
<i>Рисунок 12. Дополнительные настройки</i>	<i>16</i>
<i>Рисунок 13. Выбор типа подписи и сертификата</i>	<i>17</i>
<i>Рисунок 14. Результаты проверки ЭП</i>	<i>18</i>
<i>Рисунок 15. Завершение работы мастера при проверке ЭП</i>	<i>19</i>
<i>Рисунок 16. Настройка JavaScript в Adobe Acrobat</i>	<i>23</i>
<i>Рисунок 17. Запуск отладчика JavaScript</i>	<i>25</i>
<i>Рисунок 18. Отладчик JavaScript</i>	<i>26</i>
<i>Рисунок 19. Настройка оформления подписи</i>	<i>34</i>
<i>Рисунок 20. Пример настраиваемого оформления подписи</i>	<i>37</i>

6. Перечень таблиц

Таблица 1. Описание параметров команды <i>sign</i>	20
Таблица 2. Описание параметров команды <i>verify</i>	21
Таблица 3. Краткий обзор функций JavaScript для работы с КриптоПро PDF	27
Таблица 4. Краткий обзор объектов JavaScript для работы с КриптоПро PDF.....	28
Таблица 5. Параметры функции <i>CPSign</i>	30
Таблица 6. Параметры функции <i>CPVerify</i>	31
Таблица 7. Описание объекта <i>appearance</i>	33
Таблица 8. Описание объекта <i>customAppearance</i>	35
Таблица 9. Описание объекта <i>textField</i>	35