

# Описание протокола

<b>Вызов процедуры.</b>	<b>3</b>
<b>Таблицы. Чтение таблицы.</b>	<b>5</b>
Изменение таблицы.	6
Изменение строки многострочной таблицы.	7
Перезапись содержимого таблицы. (только для таблицы PLU)	7
Изменение строки в таблице PLU.	7
Вставка строки в таблицу. (только для таблицы PLU)	8
Удаление строки из таблицы. (только для таблицы PLU)	8
<b>Возврат ошибок.</b>	<b>8</b>
<b>Список таблиц кассы.</b>	<b>9</b>
<b>Печать чеков.</b>	<b>10</b>
Текстовый формат чека.	11
<b>Формат чека JSON.</b>	<b>15</b>
<b>Чтение и синхронизация чековой ленты.</b>	<b>18</b>
Формат ленты.	19
Представление строк чеков.	19
Определение степени заполненности электронной ленты.	21
<b>Идентификация устройства.</b>	<b>22</b>
<b>Регистрация по IP(ускоряе выполнение запросов в несоклько раз).</b>	<b>22</b>
<b>Регистрация пользователя и список пользователей.</b>	<b>22</b>
<b>Фискальная память.</b>	<b>24</b>
Процедура фискализации.	24
Таблица Hdr	24
Таблица Tax	24
Таблица Fsk	24
Чтение информации из фискальной памяти. Таблица FDay.	24
Поля таблицы FDay.	25
Поля таблицы FTax.	25
<b>Настройка сети TCP/IP.</b>	<b>25</b>
<b>Настройка видов оплат.</b>	<b>26</b>
<b>Настройка внешнего вида чека.</b>	<b>26</b>
Таблицы Hdr/Ftr.	26
Логотип.	26
Настройка принтера.	26
<b>Печать и получение отчетов.</b>	<b>27</b>
<b>Приложение 1. Список ошибок TitanXCheck.</b>	<b>29</b>
<b>Приложение 2. Список ошибок кассы.</b>	<b>30</b>

## Вызов процедуры.

Процедура вызывается методом **GET**. Имя ресурса для вызова процедуры имеет вид:  
/cgi/proc/**name?param1&param2&...&paramN**

где:

proc - имя процедуры

paramX - значение параметра процедуры.

Ниже приведены форматы параметров различных типов:

Тип	Название	Формат
<b>Двоичная информация</b>	Binary	<b>x</b> с набором шестнадцатеричных цифр Пример: <b>x010203aaBC</b>
<b>Число</b>	Number	Набор цифр с необязательным знаком минус. Дробная часть, если она нужна, пишется через точку. Примеры: <b>12345</b> , <b>-2345</b> , <b>12.5</b>
<b>Дата</b>	Date	YYYY-MM-DD
<b>Время</b>	Time	[T]HH:MM[:SS]
<b>Дата и время</b>	ISODate	YYYY[-MM[-DD]][T]HH[:MM[:SS]]]
<b>Текст</b>	Text	В формате <a href="#">кодирования URL</a> <a href="http://en.wikipedia.org/wiki/Percent-encoding">http://en.wikipedia.org/wiki/Percent-encoding</a> Пример: <b>ABC%20%E0%B1</b>

**getfmroom** – состояние фискальной памяти

Входящие параметры: Нет

Выходящие параметры:

Описание	Тип	Значения
Общее количество записей	Number	
Свободно записей	Number	

**getjrnroom**– состояние электронного журнала

Входящие параметры: Нет

Выходящие параметры:

Описание	Тип	Значения
Общее количество записей	Number	
Свободно записей	Number	

**lastreceipt**– информация по последнем зарегистрированном документе

Входящие параметры: Нет

Выходящие параметры:

Описание	Тип	Значения
Общая сумма документа	Number	
Номер документа	Number	
Дата и время документа	Number	
Номер оператора	Number	

**printfmreport** – печать отчета с БЭП

Входящие параметры:

Описание	Тип	Значения
Тип отчета	Number	1 – полный по датам 2 – полный по номера

		3 – сокращенный по датам 4 – сокращенный по номерам
Начальная дата диапазона	Date	
Конечная дата диапазона	Date	
Начальный номер отчета	Number	
Конечный номер отчета	Number	

Выходящие параметры: Нет

**printreport** – печать отчета

Входящие параметры:

Описание	Тип	Значения
Тип отчета	Number	0 – дневной обнуляющий отчет (Z1) 1 – обнуление электронного журнала 2 – печать контрольной ленты с обнулением 3 – печать отчета состояния СКНО 4 – печать отчета по открытым ресторанным заказам с автоматическим закрытием наличными 5 – печать отчета по открытым ресторанным заказам 10, 100 – дневной отчет без обнуления (X1) 20 – Отчет по проданным товарам 21 – Отчет по проданным товарам с обнулением этого отчета 102 – отчет по денежному ящику 107 – статистика продаж по времени 201 – 216 – статистика продаж по кассирам

Выходящие параметры: Нет

**puthdrfm** – перерегистрация

Входящие параметры: Нет.

Выходящие параметры: Нет.

**puttaxfm** – запись налоговых ставок в БЭП

Входящие параметры: Нет.

Выходящие параметры: Нет.

**setclock** – установка времени

Входящие параметры:

Описание	Тип	Значения
Дата и время	ISODate	

Процедура может вызываться только в случае замкнутой перемычки SA2 на системной плате или когда аппарат находится в состоянии ошибки часов.

Выходящие параметры: Нет.

**sknstate** – состояние модуля СКНО

Входящие параметры: Нет

Выходящие параметры:

Описание	Тип	Значения
Битовая маска состояния	Number	

**sound** – звуковой сигнал

Входящие параметры:

Описание	Тип	Значения
Длительность (мс)	Number	
Частота (Гц)	Number	60-4096

Выходящие параметры: Нет.

**state**—состояние текущего документа

Входящие параметры: Нет

Выходящие параметры:

Описание	Тип	Значения
Битовая маска состояния документа	Number	0x0001 – открыт фискальный чек 0x0002 – открыт не фискальный чек 0x0004 – для закрытия чека требуется оплата 0x0008 – напечатаны итоги по чеку 0x0010 – напечатан заголовок чека 0x0020 – в чеке были продажи товара 0x0040 – в чеке были вносы/выносы 0x0080 – начата расплата по чеку 0x0100 – печать контрольной ленты 0x0200 – печать служебного отчета 0x0400 – отмена документа 0x0800 – в чеке есть операции 0x1000 – начат чек возврата 0x2000 – запись занесена в ленту
Номер оператора, начавшего документ	Number	
Начальная запись документа в журнале	Number	
Начальная запись отчета в журнале	Number	

Пример вызова процедуры:

GET

/cgi/proc/sound?1000&1000

## Таблицы. Чтение таблицы.

Таблица представляет собой одну или несколько строк с несколькими полями. Как правило все таблицы аппарата имеют фиксированное количество строк. Вставка и удаление строк в таких таблицах невозможна. Однострочные таблицы представляются как объекты JSON. Многострочные - как массивы объектов.

В многострочных таблицах одно из полей является ключевым, По этому полю происходит изменение строки. В большинстве случаев это поле имеет имя **id**. В случаях, когда поле имеет другое имя, это оговаривается дополнительно в описании конкретной таблицы.

Чтение таблицы производится методом GET ресурса /cgi/tbl/name, где name - имя таблицы.

Пример чтения однострочной таблицы:

GET  
</cgi/tbl/Flg>

Ответ:

```
{"Lang":0,"AutoOff":0,"PrintOff":0,"Flg1":8448,"Flg2":0,"Feed":7,"Flg3":4}
```

Пример чтения многострочной таблицы:

GET  
</cgi/tbl/Pay>

Ответ:

```
[  
{"id":1,"Param":35,"Name":"I"},  
{"id":2,"Param":1,"Name":""},  
{"id":3,"Param":17,"Name":""},  
{"id":4,"Param":17,"Name":""}]
```

Изменение таблицы.

Изменение содержимого таблицы **/cgi/tbl/name** производится методом **PATCH**. Для изменения многострочной таблицы в изменяемых строках обязательно нужно указывать ключевое поле. Если изменение затрагивает одну строку таблицы, измененная строка возвращается в ответе при успешной модификации. При успешной модификации нескольких строк таблицы возвращается пустой объект. При ошибке модификации строки, возвращается ошибка с идентификатором этой строки, модификация дальнейших строк не производится. Формат возврата ошибок приводится в соответствующем разделе ниже. Пример модификации однострочной таблицы.

PATCH  
</cgi/tbl/Flg>  
{"Feed":1}

Ответ:

```
{"Lang":0,"AutoOff":0,"PrintOff":0,"Flg1":8448,"Flg2":0,"Feed":7,"Flg3":4}
```

Пример модификации нескольких строк многострочной таблицы

PATCH  
</cgi/tbl/Pay>  
[{"id":3,"Param":16},{ "id":4,"Param":16}]

Пример модификации одной строки многострочной таблицы

```
PATCH
/cgi/tbl/Pay
[{"id":3,"Param":17}]
```

Ответ:

```
{"id":3,"Param":17,"Name":""}
```

Пример модификации таблицы с ошибкой

```
PATCH
/cgi/tbl/
[{"id":4,"Parm":17}]
```

Ответ:

```
{"id":4,"err":{"e":"xC2"}}
```

Изменение строки многострочной таблицы.

Изменение строки таблицы по адресу **/cgi/tbl/name/id** производится методом **PATCH**.

Пример изменения строки

```
PATCH
/cgi/tbl/Pay/
{"Param":17}
```

Ответ:

```
{"id":4,"Param":17,"Name":""}
```

Перезапись содержимого таблицы. (только для таблицы PLU)

Полное стирание таблицы и вставка в неё переданной информации. Производится методом PUT по адресу /cgi/tbl/PLU.

Пример перезаписи таблицы

```
POST
/cgi/tbl/Pay/
[{"Code":1,"Name":"tov1","Price":1,"Dep":1,"Grp":1,"Tax":1,"Qty":0,"Flg":0},{ "Code":2,"Name"
:"tov2","Price":2,"Dep":1,"Grp":1,"Tax":1,"Qty":0,"Flg":0}]
```

Изменение строки в таблице PLU.

Изменение параметров товара производится методом PATCH.

```
PATCH
/cgi/tbl/PLU
```

```
[{"Code":1,"Name":"Товар изменен","Price":222,"Dep":1,"Grp":1,"Tax":1,"Qty":0,"Flg":0}]
```

Вставка строки в таблицу. (только для таблицы PLU)

Вставка в таблицу переданной информации производится методом POST по адресу /cgi/tbl/PLU.

Пример вставки строк в таблицу

POST

/cgi/tbl/PLU

```
[{"Code":3,"Name":"tov3","Price":1,"Dep":1,"Grp":1,"Tax":1,"Qty":0,"Flg":0},{ "Code":4,"Name":
:"tov4","Price":2,"Dep":1,"Grp":1,"Tax":1,"Qty":0,"Flg":0}]
```

Удаление строки из таблицы. (только для таблицы PLU)

Производится методом DELETE по адресу /cgi/tbl/PLU/code.

Пример удаления строк таблицы

DELETE

/cgi/tbl/PLU/2

## Возврат ошибок.

При возникновении ошибки операция протокола возвращает объект, в котором присутствует поле **err**.

```
{"err":...}
```

В простейшем случае это поле содержит текстовую строку с сообщением об ошибке или с кодом сообщения об ошибке. Код сообщения об ошибке имеет вид "xNN", где N - hex цифра. Список ошибок кассы вынесен в приложение 1.

Ресурс **/desc** содержит информацию по локализации, в том числе и локализованную версию сообщений об ошибках на разных языках. В настоящий момент поддерживаются украинская (поле **uk**), английская (поле **en**) и русская (поле **ru**) локализации. В поле **err**, соответствующего объекта содержатся текстовые сообщения, передаваемые кассой и их перевод на выбранный язык. Место сообщений об ошибках в ресурсе **/desc** показано ниже:

```
{
  ...
  "en":{
    ...
  },
  "uk":{
    ...
  },
  "ru":{
    ...
    "err": {
      ...
    }
  }
}
```



```

        "x3A": "Попытка модификации поля только для чтения",
        "x3B": "Неверное значение поля",
        "x3C": "Товар уже существует",
        "x3D": "По товару были продажи",
        ...
    },
    ...
},
...
}

```

В ошибке кроме кода или текста могут передаваться параметры. В этом случае поле **err** представляет собой объект с полями **e** (текст ошибки) и **p** (параметры ошибки). Например, {"err":{"e":"Error %i", "p":[100]}} представляет собой текст ошибки "Error 100". Если ошибка связана с параметром процедуры, номер параметра может передаваться в поле **f** объекта ошибки. Например: {"err":{"e":"x25", "f":2}}. Если ошибка связана с определенным полем строки таблицы, имя этого поля может передаваться в поле **f** объекта ошибки. Например: {"err":{"e" : "x25", "f":"tax"}}. Если ошибка связана с определенной строкой чека, номер этой строки может передаваться в поле **line** объекта ошибки. Например: {"err":{"e" : "x25", "line":5}}. Если вызов процедуры или другая операция генерирует несколько ошибок, полю **err** может быть присвоен массив ошибок. Например:

```
{ "err": [ { "e": "x25"}, { "e" : "x26" } ] }
```

## Список таблиц кассы.

Список таблиц кассы можно прочитать используя метод GET из ресурса **/cgi/tbl**. Список таблиц является JSON массивом объектов, каждый из которых описывает одну таблицу и содержит поля:

Поле	Описание
<b>id</b>	имя ресурса. Таблица доступна по адресу <b>/cgi/tbl/имя ресурса</b>
<b>tbl</b>	Необязательное поле. Если присутствует и не нулевое, таблица представляется JSON массивом. Иначе - JSON объектом.
<b>prefix</b>	Необязательное поле. Имя группы, для логически связанных таблиц.
<b>insertable</b>	Необязательное поле. Если присутствует и ненулевое, в таблицу можно вставлять и удалять строки.
<b>key</b>	Необязательное поле. Если присутствует, то его значение является именем ключевого поля для строк таблицы. Если отсутствует, имя ключевого поля "id"
<b>elems</b>	Массив с описанием колонок таблицы.

Описание колонки является JSON объектом с полями:

Поле	Описание
<b>name</b>	Имя поля
<b>type</b>	Тип поля. Возможные типы: "text", "number", "date", "datetime", "datetime-local", "email", "password", "time", "url", "percent", "select-one", "select-multiple", "checkbox", "radio", "range"
<b>pattern</b>	регулярный шаблон или имя шаблона в поле regex ресурса <b>/desc</b>
<b>editable</b>	Если присутствует и равно 0 - поле не редактируется
<b>renderable</b>	Если присутствует и равно 0 - поле скрытое
<b>labels</b>	метки для типов "select-one", "select-multiple", "checkbox", "radio". Могут быть

	в формате: <ul style="list-style-type: none"> <li>• строки "label1 label2 "complex label"",</li> <li>• массива строк ["label1","label2","complex label"]</li> <li>• массива объектов [{"value":1,"label":"label1"}, {"value":2,"label":"label2"}, {"value":3,"label":"complex label"}]</li> </ul>
<b>min</b>	минимальное значение числа или даты
<b>max</b>	максимальное значение числа или даты
<b>step</b>	шаг изменения числа или даты

## Печать чеков.

Чек, который должен распечатать регистратор, формируется в виде текста или в формате JSON. Этот текст методом **PATCH** отправляется на ресурс **/cgi/chk**.

После приема чека регистратор проводит следующие действия:

Вариант работы с модулем СКНО:

1. Проверяет наличие бумаги. При её отсутствии возвращается ошибка и дальнейшая обработка чека не производится. Наличие возможности печати чека можно проверить чтением ресурса **/cgi/state** позволяет получить объект текущего состояния регистратора. Для успешной печати чека поле **FskMode** объекта состояния должно быть равно **1** и поле **err** должно быть пустым.
2. Проводятся проверки на возможность работы с модулем СКНО.
3. Анализирует чек на наличие ошибок. При их наличии, ошибка возвращается пользователю, и чек не печатается.
4. Посылает документ в модуль СКНО. После получения ответа, запрещающего работу, чек не печатается.
5. Печатает чек. При возникновении ошибки печати чек оканчивается успешно. Возвращается JSON содержимое чека. Объект чека имеет поле **pending** со значением 1. При последующем устранении ошибки печати чек печатается заново. При необходимости после окончания печати кассир может выдать копию чека.
6. После успешного окончания чека возвращается JSON содержимое чека.

Вариант работы без модуля СКНО:

1. Проверяет наличие бумаги. При её отсутствии возвращается ошибка и дальнейшая обработка чека не производится. Наличие возможности печати чека можно проверить чтением ресурса **/cgi/state** позволяет получить объект текущего состояния регистратора. Для успешной печати чека поле **FskMode** объекта состояния должно быть равно **1** и поле **err** должно быть пустым.
2. Анализирует чек на наличие ошибок. При их наличии, ошибка возвращается пользователю, и чек не печатается.
3. Начинает печать чека. При возникновении ошибок печати – возвращается ошибка отсутствия бумаги, чек прекращает печататься и отменяется.
4. Печатает фискальное окончание чека (номер чека, серийный номер, фискальный логотип). При возникновении ошибки печати чек оканчивается успешно. Возвращается JSON содержимое чека. Объект чека имеет поле **pending** со значением 1. При последующем устранении ошибки печати чек допечатывается и

печатается квитанция о прерывании печати. При необходимости после окончания печати кассир может выдать копию чека.

7. После успешного окончания чека возвращается JSON содержимое чека.

#### Текстовый формат чека.

Первая строка документа может определять вид распечатанного чека.

Содержимое строки	Вид чека
<b>(F)</b>	фискальный чек
<b>(R)</b>	возвратный чек
<b>(I)</b>	чек внесения-изъятия денег из сейфа
<b>(V)</b>	чек аннулирования
<b>(P)</b>	не фискальный чек
<b>(L)</b>	копия чека
<b>(0)</b>	пустой чек

если в первой строке документа не найден тип чека, этот чек считается фискальным.

#### Строка продажи.

Для чеков типа **F, R**.

**[Количество\*]Цена "Название" Код [[Налог] [[Секция] [[Группа] [Тип кода]]]]**

Поле	Описание
<b>Количество</b>	Необязательное поле. Должно быть цифрой с не более чем тремя знаками после десятичной точки. Если поле пропущено, количество принимается равным 1.
<b>Цена</b>	Должно быть цифрой с не более чем двумя знаками после десятичной точки.
<b>Название</b>	Название товара. Если название содержит символ <b>"</b> , он удваивается.
<b>Код</b>	Код товара из не более чем 13 цифр
<b>Налог</b>	Необязательное поле. Номер налоговой ставки из таблицы <b>Tax</b> . 0 - без налога. При отсутствии поля продажа происходит с первой налоговой ставкой из таблицы налогов.
<b>Секция</b>	Необязательное поле. Номер секции из таблицы <b>Dep</b> . При отсутствии поля продажа происходит в первую секцию.
<b>Группа</b>	Необязательное поле. Номер группы из таблицы <b>Grp</b> . При отсутствии поля продажа происходит в первую группу.
<b>Тип кода</b>	Необязательное поле. Тип кода товара (0 – без EAN, 1 – EAN, 2 - услуга). При отсутствии поля продажа происходит с типом 1.

Пример:

PATCH  
 /cgi/chk  
 (F)  
 11 "Пробный товар 1" 1234567890121  
 -5  
 12 "Пробный товар 2" 1234567890122  
 -2%  
 53 "Пробный товар 3" 1234567890123  
 +10%  
 54 "Пробный товар 4" 1234567890124  
 55 "Пробный товар 5" 1234567890125  
 -50  
 56 "Пробный товар 6" 1234567890126  
 -50%  
 51 "Пробный товар 10" 1234567890111  
 +25%  
 +=15%  
 \$ 5000

Строка скидки.

Для чеков типа **F, R**.

**+/-[=] число%[;номер скидки]**

Символ	Значение
<b>+</b>	Наценка
<b>-</b>	Скидка
<b>=</b>	на промежуточный итог
<b>%</b>	Процентная
<b>номер скидки</b>	номер подтверждения льготной скидки. (Согласно требованиям НацБанка к системе передачи данных в ДПА)

Пример	Описание
<b>-10</b>	Скидка 10 грн на предыдущий товар
<b>-2.5%</b>	Скидка 2.5% на предыдущий товар
<b>+=20</b>	Наценка 20 грн на промежуточный итог
<b>+=5.5%</b>	Наценка 5.5% на промежуточный итог

Строка оплаты.

**\$(сумма)[;[номер оплаты];[номер транзакции];[номер карты]]]**

Поле	Описание
<b>Сумма</b>	Сумма оплаты. Если сумма отсутствует, считается, что она соответствует сумме всего чека.
<b>Номер оплаты</b>	Номер строки в таблице <b>Pay</b> , который соответствует типу выбранной оплаты. При отсутствии происходит оплата наличными.
<b>Номер транзакции</b>	Номер транзакции (RRN) при осуществлении оплаты через банковский терминал. (Согласно требованиям НацБанка к системе передачи данных в ДПА)
<b>Номер карты</b>	Номер банковской карты при оплате через терминал

Строка внесения-изъятия денег.

Для чеков типа **I**.

### \$[-]сумма[;номер оплаты]

Только для типа чека (IO).

Поле	Описание
-	При наличии операция является операцией изъятия. При отсутствии операция является операцией внесения.
<b>Сумма</b>	Сумма внесения или изъятия.
<b>Номер оплаты</b>	Номер строки в таблице <b>Pay</b> , который соответствует типу выбранной оплаты. При отсутствии производится внесение или изъятие наличных.

### Строка текстового комментария.

Для чеков типа **F, R, P** и **I**.

Для чеков типа **F, R** и **I** комментарии, по желанию пользователя, могут вноситься или не вноситься в электронную ленту продаж. Строки комментариев, внесенные на ленту продаж, будут печататься в копии чека и отправляться в копии чека в ГНА. Строки комментариев, не внесенные в ленту, будут распечатаны только на оригинале чека.

Для нефискальных чеков никакие комментарии в ленту продаж не вносятся.

Строка комментариев, которая будет внесена в ленту продаж, должна начинаться символом **#**.

### #строка комментария, сохраняемая в ленту

любая другая строка чека, не начинающаяся с символов **0-9+-x\$%<**, является строкой комментариев, не вносимых в ленту. Если комментарий является самой первой строкой фискального чека, его нельзя начинать также символами **{(**.

Строка, начинающаяся с **"**, всегда является комментарием, не вносимым в ленту.

### "строка комментария, не вносимая в ленту

Первый символ, который является признаком комментария, не выводится на чек.

### Печать штрих-кода.

Строка, начинающаяся с **<** считается строкой содержащей информацию о штрихкоде.

Формат строки:

**<"код" [тип [ширина [высота [отступ]]]]**

Поле	Описание
<b>Код</b>	Текст штрих-кода 1-24 символа
<b>Тип</b>	Тип штрих-кода 1 – EAN-13 (по умолчанию) 2 – CODE-128 3 – CODE-39
<b>Ширина</b>	Ширина символа в пикселях 2(по умолчанию),3,4
<b>Высота</b>	Высота штрих-кода в пикселях 1-150. По умолчанию 60
<b>Отступ</b>	Отступ до и после штрих-кода в пикселях 0-30. По умолчанию 20.

Если код выходит за границы печатаемой области, чек отменяется с ошибкой **xС2** (Неверный параметр у процедуры).

### Аннулирование документа.

Строка, начинающаяся с **>** считается строкой содержащей информацию о номере аннулируемого документа. Формат строки:

**>номер**

Поле	Описание
<b>номер</b>	Номер аннулируемого документа

#### Коррекция документа.

Строка, начинающаяся с @ считается строкой содержащей информацию о коде отменяемого товара. Отменяется первая с начала чека продажа товара с указанным кодом (если код не указан, то отменяется последняя операция). Формат строки:

**@[код]**

Поле	Описание
код	Код отменяемого товара

## Формат чека JSON.

Чек представляет собой объект с одним из полей:

Имя поля	Вид чека
<b>F</b>	фискальный чек
<b>R</b>	возвратный чек
<b>IO</b>	чек внесения-изъятия денег из сейфа
<b>VD</b>	чек аннулирования
<b>P</b>	нефискальный чек
<b>L</b>	копия чека или ресторанного счета
<b>RO</b>	ресторанный заказ
<b>VB</b>	отмена ресторанного заказа

Содержимое поля представляет собой массив строк чека. Пустой чек печатается последовательностью `{ }`. Регистратор ожидает символ `{` в качестве первого символа данных POST для чеков в формате JSON. Иначе данные интерпретируются как данные в текстовом формате.

Строка чека представляет собой объект с одним из полей:

Имя поля	Вид строки	Может находиться в чеках с типом
<b>S</b>	Строка продажи	F,R,RO
<b>D</b>	Строка скидки	F,R,RO
<b>P</b>	Строка оплаты	F,R
<b>IO</b>	Строка служебного внесения/изъятия средств	IO
<b>C</b>	Строка текстового комментария	F,R,IO,P,RO
<b>N</b>	Строка нефискального комментария	F,R,IO,P,RO
<b>BC</b>	Штрих-код	F,R,IO,P,RO
<b>VD</b>	Аннулирование документа	VD
<b>VP</b>	Коррекция документа	F,R,RO
<b>OB</b>	Открытие нового счета	RO
<b>NO</b>	Открытие нового заказа	RO
<b>CB</b>	Закрытые счета	F
<b>VB</b>	Отмена счета	VB

### Строка продажи.

Значение поля **S** есть объект с полями:

Поле	Описание
<b>qty</b>	Необязательное поле. Должно быть цифрой с не более чем тремя знаками после десятичной точки. Если поле пропущено, количество принимается равным 1.
<b>price</b>	Должно быть цифрой с не более чем двумя знаками после десятичной точки.
<b>name</b>	Название товара.
<b>code</b>	Код товара из не более чем 13 цифр
<b>tax</b>	Необязательное поле. Номер налоговой ставки из таблицы <b>Tax</b> . 0 - без налога. При отсутствии поля продажа происходит с первой налоговой ставкой из таблицы налогов.
<b>ctype</b>	Необязательное поле. Тип кода товара (0 – без EAN, 1 – EAN, 2 - услуга). При отсутствии поля продажа происходит с типом 1.

<b>dep</b>	Необязательное поле. Номер секции из таблицы <b>Dep</b> . При отсутствии поля продажа происходит в первую секцию.
<b>grp</b>	Необязательное поле. Номер группы из таблицы <b>Grp</b> . При отсутствии поля продажа происходит в первую группу.

#### Строка скидки.

Значение поля **D** есть объект с полями:

Поле	Описание
<b>sum</b>	сумма скидки или наценки. Положительное число описывает наценку. Отрицательное - скидку. Если поле присутствует в объекте, значит, строка чека является скидкой на сумму. Для процентных скидок поле должно отсутствовать.
<b>prc</b>	процент скидки или наценки. Положительное число описывает наценку. Отрицательное - скидку. Поле игнорируется, если объект имеет поле <b>sum</b> .
<b>all</b>	если присутствует и ненулевое - скидка производится на промежуточный итог.
<b>dn</b>	номер подтверждения льготной скидки. (Согласно требованиям НацБанка к системе передачи данных в ДПА)

#### Строка оплаты.

Значение поля **P** есть объект с полями:

Поле	Описание
<b>sum</b>	Сумма оплаты. Если сумма отсутствует, считается, что она соответствует сумме всего чека.
<b>no</b>	Номер строки в таблице <b>Pay</b> , который соответствует типу выбранной оплаты. При отсутствии, происходит оплата наличными.

#### Строка внесения-изъятия денег.

Значение поля **IO** есть объект с полями:

Поле	Описание
<b>type</b>	"IO" - строка описывает служебное движение денег
<b>sum</b>	Сумма внесения или изъятия. Сумма внесения должна быть положительна. Изъятия - отрицательна.
<b>no</b>	Номер строки в таблице <b>Pay</b> , который соответствует типу выбранной оплаты. При отсутствии производится внесение или изъятие наличных.

#### Строка текстового комментария.

Значение поля **C** есть объект с полями:

Поле	Описание
<b>Cm</b>	Строка текстового комментария.

#### Строка нефискального комментария.

Нефискальные комментарии не вносятся в ленту продаж. Строки комментариев, внесенные на ленту продаж (**C**), будут печататься в копии чека и отправляться в копии чека в ГНА. Строки комментариев, не внесенные в ленту (**N**), будут распечатаны только на оригинале чека.

Значение поля **N** есть объект с полями:

Поле	Описание
<b>Cm</b>	Строка нефискального комментария.
<b>Attr</b>	"w" – символы двойной ширины "h" – символы двойной высоты



	" " – символы половинной высоты Допускаются комбинации "wh" и "wl".
--	--

#### Печать штрих-кода.

Значение поля **BC** есть объект с полями:

Поле	Описание
<b>code</b>	Текст штрих-кода 1-24 символа
<b>type</b>	Тип штрих-кода 1 – EAN-13 (по умолчанию) 2 – CODE-128 3 – CODE-39
<b>width</b>	Ширина символа в пикселях 2(по умолчанию),3,4
<b>height</b>	Высота штрих-кода в пикселях 1-150. По умолчанию 60
<b>feed</b>	Отступ до и после штрих-кода в пикселях 0-30. По умолчанию 20.

Если код выходит за границы печатаемой области, чек отменяется с ошибкой xC2 (Неверный параметр у процедуры).

#### Строка аннулирования документа.

Значение поля **VD** есть объект с полями:

Поле	Описание
<b>no</b>	Номер аннулируемого документа.

#### Коррекция документа.

Значение поля **VP** есть объект с полями:

Поле	Описание
<b>code</b>	Код товара (не более чем 13 цифр). Отменяется первая с начала чека продажа товара с указанным кодом (если код не указан, то отменяется последняя операция)

#### Копия документа или ресторанного счета.

Значение поля **L** есть объект с одним из полей

Поле	Описание
<b>bill</b>	Номер открытого счета для копии(1...9999)
<b>no</b>	Номер документа

#### Открытие ресторанного счета.

Значение поля **NB** есть объект с полями:

Поле	Описание
<b>table</b>	Номер столика (1...65000).
<b>place</b>	Номер места за столиком (1...250, по умолчанию 1)

#### Новый ресторанный заказ.

Значение поля **NO** есть объект с полями:

Поле	Описание
<b>bill</b>	Номер счета (1...9999)

#### Отмена ресторанного счета

Значение поля **VB** есть объект с полями:

Поле	Описание
<b>bill</b>	Номер счета (1...9999)

#### Заккрытие ресторанный счета

Значение поля **CB** есть объект с полями:

Поле	Описание
<b>bill</b>	Номер счета (1...9999)

#### Примеры чеков в формате JSON.

Чек продажи

```
{
  "F": [
    {
      "C": {
        "cm": "Кассир: Светлана"
      }
    },
    {
      "S": {
        "code": "1",
        "price": "5",
        "name": "Конфета"
      }
    },
    {
      "S": {
        "code": "2",
        "price": "15",
        "name": "Печенье",
        "qty": "0.5"
      }
    },
    {
      "D": {
        "prc": "5",
        "all": "1"
      }
    },
    {
      "P": {}
    }
  ]
}
```

Чек служебного внесения денег

```
{
  "IO": [
    {
      "C": {
        "cm": "Кассир: Светлана"
      }
    },
    {
      "IO": {
        "sum": "120"
      }
    },
    {
      "IO": {
        "sum": "-140",
        "no": "2"
      }
    }
  ]
},
{
  "P": [
    {
      "C": {
        "cm": "Line 1"
      }
    },
    {
      "C": {
        "cm": "Line 2",
        "attr": "w"
      }
    },
    {
      "C": {
        "cm": "Line 3"
      }
    }
  ]
},
{
  "P": [
    {
      "N": {
        "cm": "Line 1"
      }
    },
    {
      "N": {
        "cm": "Line 2"
      }
    },
    {
      "N": {
        "cm": "Line 3"
      }
    }
  ]
}
```

Открытие ресторанный счета

```
{
  "RO": [
    {
      "NB": {
        "table": "1",
        "place": "1"
      }
    },
    {
      "S": {
        "code": "1",
        "price": "5",
        "name": "Конфета"
      }
    },
    {
      "S": {
        "code": "2",
        "price": "15",
        "name": "Печенье",
        "qty": "0.5"
      }
    }
  ]
}
```

Новый ресторанный заказ

```
{
  "RO": [
    {
      "NO": {
        "bill": "1"
      }
    },
    {
      "S": {
        "code": "3",
        "price": "12",
        "name": "Кофе"
      }
    }
  ]
}
```

Отмена ресторанный счета

```
{
  "VB": [
    {
      "VB": {
        "bill": "1"
      }
    }
  ]
}
```

Коррекция ресторанных заказов

Новый ресторанный заказ

```
{
  "RO": [
    {
      "NO": {
        "bill": "1"
      }
    },
    {
      "VP": {
        "code": "2"
      }
    },
    {
      "S": {
        "code": "4",
        "price": "10",
        "name": "Чай"
      }
    }
  ]
}
```

Заккрытие ресторанный счета

```
{
  "F": [
    {
      "CB": {
        "bill": "1"
      }
    },
    {
      "P": {}
    }
  ]
}
```

#### Чтение и синхронизация чековой ленты.

Прочитать имеющуюся в аппарате чековую ленту можно методом **GET** из ресурса **/cgi/chk**.

### Формат ленты.

Чековая лента представляет собой JSON массив с объектами ленты. Эти объекты имеют следующие поля:

Поле	Описание
<b>L</b>	регистрация кассира
<b>Z1</b>	Номер дневного отчета
<b>F</b>	Строки фискального чека
<b>R</b>	строки чека возврата
<b>IO</b>	строки чека служебного движения средств
<b>VD</b>	строки чека аннулирования
<b>RO</b>	строки ресторанных заказов
<b>datetime</b>	Дата и время объекта в ленте. Время приводится в формате <a href="#">Unix-time</a> (количество секунд с начала 1970 года по Гринвичу).
<b>id</b>	уникальный идентификатор объекта в чековой ленте. Это число которое у каждого следующего объекта ленты больше, чем у предыдущего.
<b>oper_id</b>	номер оператора, связанный с объектом ленты
<b>bill</b>	номер ресторанный счета

У объектов, которые представляют чеки, имеются дополнительные поля:

Поле	Описание
<b>no</b>	Номер чека.
<b>Pending</b>	Признак того, что чек еще не напечатан. Необязательное поле. При его отсутствии считается, что чек напечатан.

### Представление строк чеков.

Строка чека это объект с одним из полей со следующими именами

Поле	Описание
<b>S</b>	строка описывает продажу
<b>D</b>	Строка описывает скидку
<b>P</b>	Строка описывает оплату
<b>IO</b>	Строка описывает операцию внесения-изъятия
<b>VD</b>	Строка описывает аннулирование документа
<b>C</b>	Строка описывает текстовый комментарий
<b>CB</b>	Строка описывает закрытие ресторанный счета
<b>NO</b>	Строка описывает новый ресторанный заказ
<b>VB</b>	Строка описывает отмененный счет

Содержимое каждого из полей это объект, детализирующий операцию

#### Строка продажи.

Поле	Описание
<b>qty</b>	Необязательное поле. Должно быть цифрой с не более чем тремя знаками после десятичной точки. Если поле пропущено, количество принимается равным 1.
<b>price</b>	Должно быть цифрой с не более чем двумя знаками после десятичной точки.
<b>name</b>	Название товара.
<b>code</b>	Код товара из не более чем 13 цифр
<b>tax</b>	Необязательное поле. Номер налоговой ставки из таблицы <b>Tax</b> . 0 - без налога. При отсутствии поля продажа происходит с первой налоговой ставкой из таблицы налогов.
<b>sum</b>	Общая сумма строки

<b>ctype</b>	Тип кода товара (0 – без EAN, 1 – EAN, 2 - услуга
<b>dep</b>	Номер секции из таблицы <b>Dep</b> .
<b>grp</b>	Номер группы из таблицы <b>Grp</b> .

#### Строка скидки.

Поле	Описание
<b>sum</b>	Общая сумма скидки или наценки. Положительное число описывает наценку. Отрицательное - скидку. Это общая сумма строки чека.
<b>prc</b>	процент скидки или наценки. Положительное число описывает наценку. Отрицательное - скидку. Если поле присутствует - скидка процентная. Если отсутствует - скидка сумовая.
<b>subt</b>	сумма на которую начисляется скидка. Если поле присутствует - скидка сделана на промежуточный итог.

#### Строка оплаты.

Поле	Описание
<b>sum</b>	Сумма оплаты.
<b>no</b>	Номер строки в таблице <b>Pay</b> , который соответствует типу выбранной оплаты.

#### Строка внесения-изъятия денег.

Только для типа чека (IO).

Поле	Описание
<b>sum</b>	Сумма внесения или изъятия. Сумма внесения должна быть положительна. Изъятия - отрицательна.
<b>no</b>	Номер строки в таблице <b>Pay</b> , который соответствует типу выбранной оплаты. При отсутствии производится внесение или изъятие наличных.

#### Строка аннулирования документа.

Только для типа чека (VD).

Поле	Описание
<b>sum</b>	Сумма аннулированного документа.
<b>no</b>	Номер аннулированного документа.

#### Строка текстового комментария.

Поле	Описание
<b>cm</b>	Строка комментария

#### Строка нового ресторанного заказа

Поле	Описание
<b>bill</b>	Номер счета
<b>table</b>	Номер столика
<b>place</b>	Номер места
<b>order</b>	Номер заказа

#### Строка закрытия ресторанного счета

Поле	Описание
<b>bill</b>	Номер счета

<b>table</b>	Номер столика
<b>place</b>	Номер места
<b>orders</b>	Количество заказов в счете

#### Строка отмены ресторанный счета

Поле	Описание
<b>bill</b>	Номер счета
<b>table</b>	Номер столика
<b>place</b>	Номер места
<b>orders</b>	Количество заказов в счете

#### Определение факта изменения чековой ленты.

Изменение чековой ленты можно обнаружить периодическим чтением состояния регистратора **/cgi/state**. Поля состояния, которые необходимо контролировать для синхронизации чековой ленты.

Название	Описание
<b>serial</b>	Заводской номер аппарата. Несколько последовательных запросов статуса должны давать одинаковое значение заводского номера. В противном случае IP адрес аппарата сменил хозяина между запросами.
<b>JrnTime</b>	Время начала чековой ленты. Если между двумя последовательными запросами это время изменилось, то в аппарате появилась новая чековая лента и ее надо считывать сначала. Если время чековой ленты равно 0 - лента пуста.
<b>chkId</b>	Идентификатор последнего объекта в ленте. Если идентификатор объекта больше чем идентификатор последнего объекта ленты и серийный номер аппарата со временем ленты совпадает с предыдущим, можно дочитать ленту.

Дочитка ленты происходит HTTP методом GET по адресу **/cgi/chk?id=\_id**, где **\_id** - номер последнего считанного объекта этой ленты. Метод вернет все объекты ленты следующие за указанным.

Прочитать определенный документ из ленты можно HTTP методом GET по адресу **/cgi/chk?no=\_num**, где **\_num** - номер необходимого документа.

#### Определение степени заполненности электронной ленты.

Электронная чековая лента имеет определенный фиксированный размер, который может зависеть от модели аппарата. Лента разделена на записи фиксированного размера. Чек разной длины занимает в ней разное число записей. Размер чека в записях можно вычислить как разницу между значениями поля **id** этого и предыдущего чека. Свободное место в ленте можно проверить вызвав процедуру **/cgi/proc/getjrnroom**. Процедура возвращает объект с полями **Total** и **Used**, в которых передается общее количество записей в ленте и использованное количество записей. Вычислив оставшееся количество записей и разделив его на размер среднего чека в ленте можно получить представление об оставшемся в ней месте. Лента очищается печатью дневного отчета (процедура **/cgi/proc/printreport?0**). В этот момент регистратор проводит следующие действия:

1. Передаются данные отчета в СКНО
2. Печатает текст отчета.
3. Переносит итоги дня в фискальную память.

## Идентификация устройства.

Информацию об устройстве можно получить, прочитав ресурс `/cgi/dev_info`. Аппарат возвращает JSON объект со следующими полями:

Поле	Описание
<b>dev_zn</b>	Заводской номер
<b>dev_ver</b>	Версия ПО
<b>dev_dat</b>	Дата прошивки устройства
<b>skno_num</b>	Номер СКНО
<b>reg_num</b>	Регистрационный номер
<b>unp_num</b>	Уникальный номер плательщика
<b>prot</b>	Версия протокола аппарата. Поле отсутствует для аппаратов, в которых номер оплаты при печати чеков на единицу меньше, чем id соответствующего вида оплаты в таблице <code>/cgi/tbl/Pay</code> Значение поле равно 1 для всех остальных случаев

## Регистрация по IP(ускоряе выполнение запросов в несколько раз).

Аппарат ведет список IP адресов, с которых авторизация к ресурсам `/cgi/*` не требуется. Список адресов находится в таблице по адресу `/cgi/tbl/whiteIP`. В таблице фиксированное число строк. Поля таблицы:

Поле	Описание
<b>id</b>	идентификатор строки
<b>ip</b>	IP адрес. При адресе 0.0.0.0 строка считается свободной.
<b>login</b>	имя оператора, который считается зарегистрированным по этому адресу.

Если таблица **whiteIP** не заполнена до конца, существует упрощенный способ добавить туда строку - вызов процедуры `/cgi/proc/register`. Вызов процедуры приводит к сохранению в свободной строке процедуры IP адреса, с которого она была вызвана, и оператора, от имени которого она была вызвана. Если IP адрес уже присутствует в таблице, номер оператора в строке с ним меняется на номер оператора, который вызвал процедуру. Процедура с параметром `/cgi/proc/register?clear` освобождает строку, если IP адрес был записан ранее.

## Регистрация пользователя и список пользователей.

Для печати чеков и редактирования программных установок аппарата имена операторов и пароли берутся из таблицы **Oper** аппарата. В качестве имени пользователя берется **номер** оператора из таблицы. Поля таблицы **Oper**:

Поле	Описание
<b>id</b>	Номер оператора
<b>Name</b>	Имя оператора
<b>Pswd</b>	Пароль оператора

При смене оператора необходимо вызвать процедуру регистрации для изменения пользователя в таблице whiteIP. Если номер вызвавшего процедуру оператора равен сохраненному в таблице, то поле **Login** освобождается, что позволяет в дальнейшем изменять оператора без вызова процедуры регистрации. В этом случае после подключения аппарата следут контролировать поле оператора по умолчанию в таблице whiteIP.

**\*\***(только для исполняемого модуля)

Если поле **Login** не содержит номер оператора и программа запущена без явного указания номера и пароля оператора, то операция будет произведена от имени оператора по умолчанию. Оператор по умолчанию: оператор № 1, пароль 1.

Пример 1:

TitanXCheck.exe "регистрация IP интерфейса в WhiteList KCA.txt" "2" "2"

GET  
/cgi/proc/register

Ответ:

```
{ "id":1, "IP":"169.254.207.82", "Login":"2"},  
{ "id":2, "IP":"0.0.0.0", "Login":""},  
{ "id":3, "IP":"0.0.0.0", "Login":""},  
{ "id":4, "IP":"0.0.0.0", "Login":""},  
{ "id":5, "IP":"0.0.0.0", "Login":""}]
```

Все последующие действия будут производиться от имени оператора №2.

Пример 2:

Повторный вызов

TitanXCheck.exe "регистрация IP интерфейса в WhiteList KCA.txt" "2" "2"

GET  
/cgi/proc/register

Ответ:

```
{ "id":1, "IP":"169.254.207.82", "Login":""},  
{ "id":2, "IP":"0.0.0.0", "Login":""},  
{ "id":3, "IP":"0.0.0.0", "Login":""},  
{ "id":4, "IP":"0.0.0.0", "Login":""},  
{ "id":5, "IP":"0.0.0.0", "Login":""}]
```

Далее без явного указания номера оператора все операции будут произведена от имени оператора по умолчанию. Оператор по умолчанию: оператор № 1.

## Фискальная память.

### Процедура фискализации.

Процедура фискализации проводится на нефискальном устройстве. Все указанные ниже операции возможны только от имени пользователя **service**. Для фискализации нужно провести следующие шаги:

Шаг	Описание
1	Установка текущего времени. Процедура <b>/cgi/proc/setclock</b> с параметром текущего времени в формате ISODate. Например: <b>/cgi/proc/setclock?2014-10-05T15:56:07</b>
2	Установка номеров СКНО, регистрационного, и уникального номера плательщика через таблицу <b>Fsk</b> .
3	Установка позиции десятичной точки в таблице <b>Fsk</b> , поле <b>NumPnt</b> .
4	Установка налоговых ставок в таблице <b>Tax</b> .
5	Вызов процедуры фискализации <b>/cgi/proc/fiscalization</b> .

### Таблица Hdr

Поле	Описание
<b>id</b>	номер строки
<b>Param</b>	1 - обычный шрифт. 2 - выделенный шрифт.
<b>Line</b>	Текст строки

В фискализированной кассе для внесения изменений в 4-5-ю строку заголовка в фискальную память необходимо вызвать процедуру **/cgi/proc/puthdrfm**.

### Таблица Tax

Поле	Описание
<b>Id</b>	номер налога
<b>Prc</b>	процент налога
<b>TaxName</b>	Название налога

В фискализированной кассе для записи налоговых ставок и положения десятичной точки (NumPnt таблица Fsk) в фискалку используется вызов процедуры **/cgi/proc/puttaxfm**.

### Таблица Fsk

Поле	Описание
<b>NumSKNO</b>	Номер СКНО
<b>RegNum</b>	Регистрационный номер
<b>UNP</b>	Уникальный номер плательщика
<b>NumPnt</b>	Позиция десятичной точки в денежных суммах
<b>NumChk</b>	Номер первого чека в открытой смене

### Чтение информации из фискальной памяти. Таблица FDay.

Прочитать информацию из фискальной памяти можно по адресу **/cgi/tbl/FDay**. Поскольку в заполненном аппарате фискальных записей может быть несколько тысяч, предусмотрен режим частичного чтения таблицы по адресу **/cgi/tbl/FDay?s=startIdx&e=endIdx**, где **startIdx** - индекс первого возвращаемого отчета, а **endIdx** - индекс отчета, следующего за последним возвращаемым. Например, команда **/cgi/tbl/FDay?s=1&e=4** возвращает отчеты 1,2,3.

Если в запросе не задан начальный индекс, он считается равным 1. Если не указан конечный индекс, он равен индексу отчета, следующему за последним. Например команда **/cgi/tbl/FDay?e=4** вернет отчеты 1,2,3. Команда **/cgi/tbl/FDay?s=4** вернет отчеты от четвертого до последнего включительно.



Команда воспринимает и отрицательные индексы. Так, -1 - это индекс последнего отчета, -2 - предпоследнего и т.д. Например, команда /cgi/tbl/FDay?s=-1 вернет только последний отчет.

#### Поля таблицы FDay.

Поле	Описание
<b>id</b>	номер записи в фискальной памяти
<b>valid</b>	Флаг действительной записи 1-действительна, 0-нет.
<b>Date</b>	Дата записи в Unix-time
<b>STax0-STax4</b>	Сумма, подлежащая налогообложению по соответствующим ставкам.
<b>SRet0-SRet4</b>	Сумма, подлежащая налогообложению в возвратных чеках.
<b>NChk</b>	Количество чеков продаж
<b>NRet</b>	Количество возвратных чеков

Строки в таблицу добавляются вызовом процедуры /cgi/proc/printreport?0

Свободное место в фискальной памяти можно проверить, вызвав процедуру /cgi/proc/getfmroom. Процедура возвращает объект с полями **Total** и **Used**, в которых передается общее количество записей в фискальной памяти и использованное количество записей.

#### Поля таблицы FTax.

Эта таблица содержит данные о налоговых ставках и времени их введения в аппарате.

Строки в таблицу добавляются вызовом процедуры /cgi/proc/puttaxfm

Поле	Описание
<b>id</b>	номер строки
<b>valid</b>	Флаг действительной записи 1-действительна, 0-нет.
<b>Date</b>	Дата записи в Unix-time
<b>NPnt</b>	Позиция десятичной точки в суммах
<b>Tax1-Tax4</b>	Ставки налогов
<b>TaxName1-TaxName4</b>	Названия налогов.
<b>NDay</b>	Номер дневного отчета, после которого записана запись.

#### Настройка сети TCP/IP.

Настройка сетевых интерфейсов аппарата производится в таблице TCP.

Поле	Описание	
<b>id</b>	Номер интерфейса.	
<b>Addr</b>	IP адрес адаптера. Значение "0.0.0.0" говорит о том, что адрес будет определяться автоматически с использованием протоколов DHCP и AutoIP.	
<b>Mask</b>	Маска сети	
<b>Gate</b>	Шлюз по умолчанию	
<b>DNS</b>	Адрес DNS сервера	
<b>MAC</b>	MAC адрес адаптера	
<b>AdptFlg</b>	Флаги адаптера	
	<b>Бит</b>	<b>Значение</b>
	<b>1</b>	разрешена скорость 10 Mbps
	<b>2</b>	разрешена скорость 100 Mbps
	<b>3</b>	разрешена скорость 1000 Mbps
	<b>4</b>	разрешить дуплекс
	<b>5</b>	автоматический режим
	<b>6</b>	разрешить определение MDI/MDI-X

### Настройка видов оплат.

Настройка видов оплат производится в таблице /cgi/tbl/Pay. Таблицу можно изменять только при пустой ленте продаж. В таблице присутствуют следующие поля:

Поле	Описание	
<b>id</b>	Порядковый номер вида оплаты	
<b>Param</b>	Флаги свойств оплаты	
	<b>Величина</b>	<b>Описание</b>
	<b>1</b>	вид оплаты разрешен
	<b>2</b>	по виду оплаты возможна сдача
	<b>16</b>	вид оплаты не лежит в сейфе
	<b>32</b>	вид оплаты может использоваться для оплаты возвратных чеков
	<b>64</b>	безналичный вид оплаты
	Значение поля является суммой необходимых флагов	
<b>Name</b>	Название вида оплаты	

### Настройка внешнего вида чека.

#### Таблицы Hdr/Ftr.

Таблицы заголовка чека (/cgi/tbl/Hdr) и таблица окончания чека (/cgi/tbl/Ftr) используются для формирования соответствующих частей чека и имеют следующую структуру:

Поле	Описание
<b>id</b>	номер строки
<b>Param</b>	1 - обычный шрифт. 2 - выделенный шрифт.
<b>Line</b>	Текст строки

Таблицы можно изменять только при пустой ленте продаж.

#### Логотип.

Ресурс /cgi/logo.bmp позволяет загрузить методом POST монохромный рисунок в формате BMP который будет использоваться в качестве графического логотипа чека. Методом GET можно получить текущий используемый логотип. Размеры загружаемого рисунка не должны превышать размеры рисунка, который возвращается кассой. Размер файла должен быть равен 2622 байта.

#### Настройка принтера.

Настройка контрастности (**PrContr**) и равномерности (**PrEqual**) печати находятся в таблице **Adm**.

Некоторые флаги, влияющие на печать, находятся в таблице **Fig**.

Поле **PrintOff**.

Флаг	Описание
<b>1</b>	выключить принтер
<b>2</b>	печать заголовка следующего чека сразу после окончания печати предыдущего
<b>4</b>	включение обрезчика
<b>8</b>	использовать полную обрезку

Значение поля является суммой используемых флагов.

Поле **Flg1**.

Флаг	Описание
<b>1</b>	Следить за запасом товара
<b>2</b>	Не используется
<b>4</b>	Следить за дробным количеством
<b>8</b>	Не используется
<b>16</b>	Сортировать товары в отчетах
<b>32</b>	Не используется
<b>64</b>	Не используется
<b>128</b>	Не используется
<b>256</b>	Разрешить продажи с 0-й ценой
<b>512</b>	Удалять после Z1 товары с 0-м запасом
<b>1024</b>	Удалять после Z1 все товары
<b>2048</b>	Сквизирование Z отчетов
<b>4096</b>	Автоматический вывод Z2 после Z1
<b>8192</b>	Не используется
<b>16384</b>	Не используется
<b>32768</b>	Не используется

Значение поля является суммой используемых флагов.

Поле **Flg3**.

Флаг	Описание
<b>1</b>	Печатать код товара в чеке
<b>2</b>	Печатать имя отдела в чеке
<b>4</b>	Печатать номер отдела в чеке
<b>8</b>	Не используется
<b>16</b>	Не выводить Z1 отчет без инкассации всех наличных денег
<b>32</b>	Не используется
<b>64</b>	Не печатать имя кассира
<b>128</b>	Не используется
<b>256</b>	Не печатать логотип пользователя
<b>512</b>	включить ресторанный режим
<b>1024</b>	Работа с заказами любым кассиром
<b>2048</b>	Не печатать копии заказов для развоза на кухню
<b>4096</b>	Не используется
<b>8192</b>	Не используется
<b>16384</b>	Не используется
<b>32768</b>	Печатать сумму налога после каждой продажи

Значение поля является суммой используемых флагов.

В поле **Feed** таблицы **Flg** указывается количество строк, которые принтер прогоняет между чеками.

## Печать и получение отчетов.

Отчеты аппарата печатаются процедурой /cgi/proc/printreport?**no**, где **no** тип отчета

Отчеты из фискальной памяти печатаются процедурой

/cgi/proc/printfmreport?**RepID&BegDate&EndDate&BegZRep&EndZRep**, где

RepID	Вид фискального отчета
-------	------------------------

	1 - полный отчет по датам 2- полный отчет по номерам Z отчетов 3 - краткий отчет по датам 4 - краткий отчет по номерам Z отчетов
BegDate EndDate	Начальная и конечная дата отчета ( RepID 1 или 3 )
BegZRep EndZRep	Начальный и конечный номер Z отчета (RepID 2 или 4)

Получить данные по остатку денег в кассе можно из ресурса /cgi/rep/pay. Возвращается массив объектов с полями **no** - номер оплаты и **sum** - остаток по этому виду оплаты. Остаток считается как разница служебных внесений и изъятий плюс сумма выторга. Для видов оплаты, которые не находятся в сейфе и для которых не имеют смысла операции служебного движения средств в качестве суммы выступает просто сумма выторга этим видом оплаты.

## Приложение 1. Список ошибок TitanXCheck.

<b>Код</b>	<b>Описание</b>
<b>1000</b>	Таймаут
<b>1001</b>	Ошибка записи данных в КСА
<b>1002</b>	Ошибка чтения файла:
<b>1003</b>	Демонстрационный режим: печать чеков недоступна
<b>1004</b>	Отключен режим терминала
<b>1005</b>	Ошибка открытия файла:
<b>1006</b>	Нет связи
<b>1007</b>	Ошибка авторизации/адрес не найден

## Приложение 2. Список ошибок кассы.

Код	Описание
<b>x01</b>	Цена не указана
<b>x02</b>	Количество не указано
<b>x03</b>	Отдел не указан
<b>x04</b>	Группа не указана
<b>x25</b>	Нет бумаги
<b>x31</b>	Пользователь уже зарегистрирован
<b>x32</b>	Неверный пароль
<b>x33</b>	Неверный номер таблицы
<b>x34</b>	Доступ к таблице запрещен
<b>x35</b>	Умолчание не найдено
<b>x36</b>	Неверный индекс
<b>x37</b>	Неверное поле
<b>x38</b>	Таблица переполнена
<b>x39</b>	Неверная длина двоичных данных
<b>x3A</b>	Попытка модификации поля только для чтения
<b>x3B</b>	Неверное значение поля
<b>x3C</b>	Товар уже существует
<b>x3D</b>	По товару были продажи
<b>x3E</b>	Запрос запрещен
<b>x3F</b>	Неверная закладка
<b>x40</b>	Ключ не найден
<b>x41</b>	Процедура уже выполняется
<b>x42</b>	Количество товара отрицательно
<b>x8A</b>	Нет бумаги для контрольной ленты
<b>x8B</b>	Нет бумаги
<b>x8D</b>	Выдача сдачи запрещена
<b>xA6</b>	Есть 3 или более непереданных отчета
<b>xBB</b>	Лента не пуста
<b>xB6</b>	Ресторанный режим не активен
<b>xB7</b>	Ресторанный счет не открыт
<b>xB8</b>	Переполнение количества заказов
<b>xB9</b>	Ресторанный чек открыт
<b>xBA</b>	Неверный номер счета
<b>xBС</b>	Режим тренировки
<b>xBD</b>	Текущая дата неверна
<b>xBE</b>	Запрещено изменение времени
<b>xBF</b>	Истек сервисный таймер
<b>xC0</b>	Ошибка работы с терминалом НСМЕП
<b>xC1</b>	Неверный номер налога
<b>xC2</b>	Неверный параметр у процедуры
<b>xC3</b>	Режим фискального принтера не активен
<b>xC4</b>	Изменялось название товара или его налог
<b>xC5</b>	СКНО занято
<b>xC6</b>	ошибка обмена с СКНО (нет связи)
<b>xC7</b>	смена не открыта
<b>xC8</b>	СКНО переполнено
<b>xC9</b>	Неверный статус СКНО
<b>xCA</b>	Ошибка идентификации СКНО
<b>xCB</b>	запрещена операция продажи

<b>xCC</b>	Начата операция возврата
<b>xCE</b>	неверный тип кода товара
<b>xCF</b>	Не выведен отчет Z1
<b>xD0</b>	Не сделана инкассация денег
<b>xD1</b>	Сейф не закрыт
<b>xD2</b>	Печать ленты прервана
<b>xD3</b>	Достигнут конец текущей смены, или изменилась дата
<b>xD4</b>	Не указано значение процентной скидки по умолчанию
<b>xD5</b>	Не указано значение скидки по умолчанию
<b>xD6</b>	Дневной отчет не выведен
<b>xD7</b>	Дневной отчет уже выведен (и пуст)
<b>xD8</b>	Нельзя отменить товар на который сделана скидка без ее предварительной отмены
<b>xD9</b>	Товар не продавался в этом чеке
<b>xDA</b>	Нечего отменять
<b>xDB</b>	Отрицательная сумма продажи товара
<b>xDC</b>	Неверный процент
<b>xDD</b>	Нет ни одной продажи
<b>xDE</b>	Скидки запрещены
<b>xDF</b>	Неверная сумма платежа
<b>xE0</b>	Тип оплаты не предполагает введения кода клиента
<b>xE1</b>	Неверная сумма платежа
<b>xE2</b>	Идет оплата чека
<b>xE3</b>	Товар закончился
<b>xE4</b>	Номер группы не может меняться
<b>xE5</b>	Неверная группа
<b>xE6</b>	Номер отдела не может меняться
<b>xE7</b>	Неверный отдел
<b>xE8</b>	Нулевое произведение количества на цену
<b>xE9</b>	Переполнение внутренних сумм
<b>xEA</b>	Дробное количество запрещено
<b>xEB</b>	Неверное количество
<b>xEC</b>	Цена не может быть изменена
<b>xED</b>	Неверная цена
<b>xEE</b>	Товар не существует
<b>xEF</b>	Начат чек внесения-изъятия денег
<b>xF0</b>	Чек содержит продажи
<b>xF1</b>	Не существующий или запрещенный тип оплаты
<b>xF2</b>	Поле в строке переполнено
<b>xF3</b>	Отрицательная сумма по дневному отчету
<b>xF4</b>	Отрицательная сумма по чеку
<b>xF5</b>	Чек переполнен
<b>xF6</b>	Дневной отчет переполнен
<b>xF7</b>	Чек для копии не найден
<b>xF8</b>	Оплата чека не завершена
<b>xF9</b>	Кассир не зарегистрирован
<b>xFA</b>	У кассира нет прав на эту операцию
<b>xFB</b>	Нефискальный чек не открыт
<b>xFC</b>	Чек не открыт
<b>xFD</b>	Нефискальный чек уже открыт
<b>xFE</b>	Чек уже открыт
<b>xFF</b>	Переполнение ленты

