[**http://www.delphin.com.ua**](http://www.delphin.com.ua)

# Описание COM компонента для кассовых аппаратов Екселлио DP-25, DP-35, DP-45.

#### Редакция: 20 августа 2015 года

Павел Болюбаш

Skype: San7A-

093 80 27 007

067 610 83 00

[5avel@hotmail.com](mailto:5avel@hotmail.com)

**Оглавление**

[Описание COM компонента для кассовых аппаратов Екселлио DP-25, DP-35, DP-45. 0](#_Toc427857374)

[Соединение с ККМ. 2](#_Toc427857375)

[Соединение по TCP. 2](#_Toc427857376)

[Соединение по RS232(СОМ порт). 2](#_Toc427857377)

[Класс Delphin.ECRfor1c 3](#_Toc427857378)

[Метод Connect 3](#_Toc427857379)

[Метод Disconnect 3](#_Toc427857380)

[Метод WriteArticul 3](#_Toc427857381)

[Метод ReadArticul 4](#_Toc427857382)

[Метод ClearArticul 4](#_Toc427857383)

# Соединение с ККМ.

Соединение с аппаратом осуществляется двумя способами, на выбор: по протоколу TCP, и по RS232 (СОМ порт).

## Соединение по TCP.

Для работы по TCP аппарату необходимо присвоить статический IP адрес, TCP порт 5999 (изменить нельзя).

Программирование(пароль 30)/Конфигурация/LAN (прописать настройки)

Также нужно настроить:

Программирование(пароль 30)/Конфигурация/Коммуникация/ПК (Нi - 0).

Если аппарат с GSM модемом то GSM модем нужно отключить.

Сервис(пароль 30)/ДПС терминал/Конфигурация/викор GPRS - нi

Для работы по TCP через интернет, аппарат является сервером и по этому у него должен быть статический IP-адрес, или роутер с настроенным Dyn-DNS клиентом. Также в случае использования роутера необходимо пробросить на IP адрес аппарата порт 5999.

## Соединение по RS232(СОМ порт).

Настройка аппарата Программирование(пароль 30)/Конфигурация/ Коммуникация/ПК (СОМ1 – 1 или СОМ2 – 2).

Перевести аппарат в режим Соединение с ПК.

Скорость по умолчанию 115200.

# Класс Delphin.ECRfor1c

Класс не содержит открытых статических методов и свойств.

Методы и свойства объекта класса Delphin.ECRfor1c:

## Метод Connect

<summary>

Метод предназначен для установки соединения с ККМ.

</summary>

<param name="ipOrCom">IP адрес аппарата(DNS имя) или номер СОМ порта, в зависиммости от параметра isRS232.

<param name="portOrSpeed">ТСР порт или сеорость СОМ порта, в зависиммости от параметра isRS232.

<param name="isRS232">true - Подключение по СОМ порту, false - подключение по TCP.

<param name="timeout"> Таймаут подключение по СОМ порту. (в милмсекундай, по умолчанию 500мс).

<returns>True - при успешном соединении, False - в случае ошибки.

**bool Connect(string ipOrCom, int portOrSpeed, ipOrCom, int portOrSpeed = 115200, bool isRS232 = true, int timeout = 500);**

## Метод Disconnect

<summary>

Метод предназначен для закрытия соединения с ККМ.

</summary>

<returns>Возвращаемое значение: логического типа, ИСТИНА, если функция выполнена успешно, ЛОЖЬ – если возникла ошибка.

**bool Disconnect()**

## Метод WriteArticul

Метод предназначен для программирования артикула в кассовый аппарат.

Максимальное количество артикулов 100 000.

**<param name="Code">**целое число, код артикула по которому будет осуществляться продажа товара.

**<param name="Name">**строка, наименование товара. Максимальная длина наименования 32 символа(включая пробелы).

**<param name="Price">** вещественное число, цена товара.

**<param name="Qnty">** вещественное число, количество товара. (значение 0)

**<param name="Dep">** целое число, номер отдела. (значение 1)

**<param name="Grp">** целое число, номер группы товара. (значение 1)

**<param name="Tax">** целое число, номер налоговой группы товара. (значение 1 налог А)

**<param name="BarCode">** строка, штрих-код товара (до 13 цифр)

**<returns>**Возвращаемое значение: логического типа, ИСТИНА, если функция выполнена успешно, ЛОЖЬ – если возникла ошибка.

**bool** **WriteArticul**(**int** Code, **string** Name, **double** Price, **double** Qnty, **byte** Dep, **byte** Grp, **byte** Tax, **string** BarCode)

## Метод ReadArticul

**bool** **ReadArticul**(**int** Code)

Метод предназначен для чтения артикула и данных о продажах по нему из РРО.

После успешного выполнения данной функции становятся доступны свойства, в которых записана

информация о прочитанном артикуле.

Свойства доступные (только для чтения) после успешного выполнения функции чтения артикула ReadArticul

ArtCode – код товара (целое число)

ArtName – название товара (строка) 32

ArtPrice – цена товара (вещественное число)

ArtQnty – количество товара (вещественное число)

ArtDep – номер отдела товара (целое число)

ArtGrp – номер группы товара (целое число)

ArtTax – номер налоговой группы товара (целое число)

ArtBarCode – штрих-код товара (строка)

ArtSaleQnty – количество проданного товара (вещественное число)

ArtSaleSum– сумма проданного товара (вещественное число)

<param name="Code">Code – целое число, Код товара. <param>

<returns>Возвращаемое значение: логического типа, ИСТИНА, если функция выполнена успешно, ЛОЖЬ – если возникла ошибка.

## Метод ClearArticul

Метод предназначен для удаления артикула запрограммированного в кассовый аппарат.

Кассовый аппарат удаляет артикул только в том случае, если по данному артикулу не было оборота (продаж). Т.е. это после снятия Z отчета.

<param name="Code"> целое число, код товара, который будет удален. </param>

<returns>Возвращаемое значение: логического типа, ИСТИНА, если функция выполнена успешно, ЛОЖЬ – если возникла ошибка. </returns>

**public** **bool** **ClearArticul**(**int** Code)

# Чтение продаж из ККМ.

Перед выполнением метода **GetCheck,** нужно установить дату загрузки продаж в переменную DataSalesTo.

Можно использовать диапазон дат с DataSalesFrom по DataSalesTo. (Формат даты **DD-MM-YY**)

**bool** **GetCheck**() Получает чек для чтения продаж.

Свойства доступные (только для чтения) после успешного выполнения функции **GetCheck.**

JCheckNum – целое число, номер фискального чека

JCheckIsReturn – ИСТИНА - Чек возврата, ЛОЖЬ - Продажный чек.

JCheckIsVoid - ИСТИНА - Чек отменен, ЛОЖЬ - не отменен.

JCheckDate - строка, дата чека DD.MM.YY<SPACE>hh:mm:ss

JCheckDis - вещественное число, процент скидки на весь Чек 0.00…99.99 (отрицательный - скидка, положительная – надбавка)

JCheckNumZRep –номер Z-отчета.

JCheckPayCount - Количество оплат максимум 3

JCheckPay1Type – тип отплаты 1 ( 0 – нал, 1 – карта, 2 - кредит)

JCheckPay1Sum – сума оплаты 1

JCheckPay2Type – тип отплаты 2 ( 0 – нал, 1 – карта, 2 - кредит)

JCheckPay2Sum – сума оплаты 2

JCheckPay3Type– тип отплаты 3 ( 0 – нал, 1 – карта, 2 - кредит)

JCheckPay3Sum – сума оплаты 3

JCheckSum – сумма чека.

JCheckTax1Sum сумма налога А.

JCheckTax1Zbir сумма сбора А.

**bool** **ReadSales**()

Читает продажи внутри чека, после успешного выполнения метода **GetCheck()**.

Свойства доступные (только для чтения) после успешного выполнения функции **ReadSales.**

JArtCode – целое число, код артикула

JArtDep – целое число, номер отдела

JArtTax – целое число, номер налоговой группы

JArtGrp – целое число, номер товарной группы

JArtVoid – логический тип, признак аннулирования товара (ЛОЖЬ -артикул продан, ИСТИНА -артикул продан, а потом аннулирован)

JArtPrice – вещественное число, цена проданного артикула (без учета скидки)

JArtQnt – вещественное число, количество проданного артикула

JArtDis– – вещественное число, процент скидки на товар 0.00…99.99 (отрицательный - скидка, положительная – надбавка)

**bool** **Beep**(**int** Tone, **int** Leng)

Подает звук на динамик ККМ.

Пример работа с товарами 1c v8:

ecr1c = новый COMОбъект("Delphin.ECRfor1C"); // Создание объекта из внешней компоненты

Сообщить(ecr1c.Connect("in4.pp.ua", 5999, Ложь)); // для ТСР

Сообщить(ecr1c.Connect("1", 115200, Истина, 500)); // для СОМ порт

ecr1c.Beep(1000, 100);

Сообщить(ecr1c.WriteArticul(20, "Test20", 99.30, 20.5, 1, 1, 1, ""));

Сообщить(ecr1c.ReadArticul(20));

Сообщить(ecr1c.ArtCode);

Сообщить(ecr1c.ArtName);

Сообщить(ecr1c.ArtPrice);

Сообщить(ecr1c.ArtQnty);

Сообщить(ecr1c.ArtDep);

Сообщить(ecr1c.ArtGrp);

Сообщить(ecr1c.ArtTax);

Сообщить(ecr1c.ArtNC);

Сообщить(ecr1c.ArtNK);

Сообщить(ecr1c.ArtBarCode);

Сообщить(ecr1c.ArtSaleQnty);

Сообщить(ecr1c.ArtSaleSum);

Сообщить(ecr1c.ClearArticul(20));

Сообщить(ecr1c.ReadArticul(20));

Сообщить(ecr1c.ArtCode);

Сообщить(ecr1c.ArtName);

Сообщить(ecr1c.ArtPrice);

Сообщить(ecr1c.ArtQnty);

Сообщить(ecr1c.ArtDep);

Сообщить(ecr1c.ArtGrp);

Сообщить(ecr1c.ArtTax);

Сообщить(ecr1c.ArtNC);

Сообщить(ecr1c.ArtNK);

Сообщить(ecr1c.ArtBarCode);

Сообщить(ecr1c.ArtSaleQnty);

Сообщить(ecr1c.ArtSaleSum);

Сообщить(ecr1c.Disconnect());

Чтение Продаж 1c v8:

ecr = новый COMОбъект("Delphin.ECRfor1C"); // Создание объекта из внешней компоненты

Сообщить(ecr.Connect("in4.pp.ua", 5999, 1));

Попытка

ecr.Beep(1000, 1000);

ecr.DataSalesFrom = "28-07-15";

ecr.DataSalesTo = "28-07-15";

Пока ecr.GetCheck() Цикл

Сообщить(Символы.ПС+ "Чек - " + ecr.JCheckNum

+ " Дата - " + ecr.JCheckDate + " Скидка на чек "

+ ecr.JCheckDis + "%" + " Возврат - " + ecr.JCheckIsReturn

+" Аннулирован - "+ecr.JCheckIsVoid);

Пока ecr.ReadSales() Цикл

Сообщить(Строка(ecr.JArtCode)+" "+СокрЛП(ecr.JArtName)

+" "+Число(ecr.JArtPrice)+" "+Строка(ecr.JArtQnt)

+" "+Строка(ecr.JArtSum)+" "+Строка(ecr.JArtDis)

+" "+Строка(ecr.JArtVoid));

КонецЦикла;

КонецЦикла;

ecr.Disconnect();

Исключение

ecr.Disconnect();

КонецПопытки;