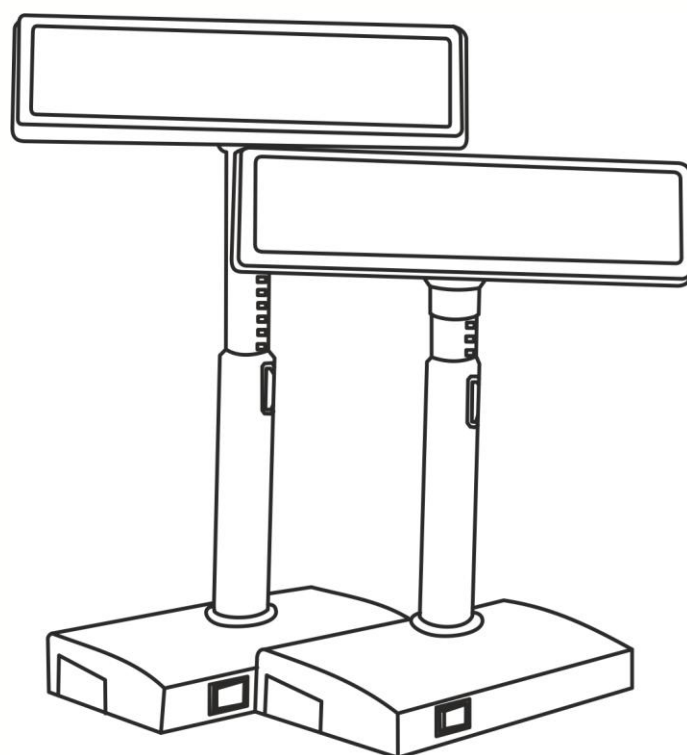


Драйвер дисплеев покупателя

v. 8

АТОН



Руководство программиста

Руководство программиста от 24.07.2014
Драйвер дисплеев покупателя v.8

Содержание

Введение	6
Сокращения	6
Условные обозначения	6
О руководстве.....	6
Что такое «Драйвер дисплеев покупателя»?	7
Интерфейс драйвера	8
Структура описания методов и свойств	9
Классификация свойств	11
Отличия от драйвера версии 6	12
Определения	12
Общие свойства.....	13
ResultCode	13
ResultDescription	14
Системные свойства	14
Version	14
ServerVersion.....	15
ApplicationHandle.....	15
DeviceDescription	15
IsDemo	15
Информационные свойства модели	15
CapBlink.....	17
CapBrightness	17
CapCharacterSet.....	17
CharacterSetList	18
CapDescriptors	19
DeviceDescriptors	19
CapHMarquee	19
CapVMarquee	20
CapICharWait.....	20
CapDownloadFonts	20
CapSelfTest	20
DeviceWindows	21

DeviceColumns.....	21
DeviceRows.....	21
GetDeviceColumns(Row).....	21
CapStatus.....	21
Логические устройства	21
Свойства текущего ЛУ	23
AddDevice () Добавить Устройство ().....	28
DeleteDevice () Удалить Устройство ()	30
ShowProperties () Показать Страницу Свойств ()	31
Методы работы с окнами и областями вывода	32
CurrentWindow	32
Columns.....	32
Rows	33
CursorColumn.....	33
CursorRow.....	33
CursorUpdate	33
CreateWindow (Yview, Xview, Hview, Wview, Hwindow, Wwindow) Создать Окно (Yобласти, Xобласти, Vобласти, Шобласти, Вокна, Шокна).....	34
DestroyWindow () Удалить Окно ()	35
AddViewPort (Yview, Xview, Hview, Wview) Добавить Область Вывода (Yобласти, Xобласти, Vобласти, Шобласти)	36
ClearViewPorts () Очистить Область Вывода ()	36
Методы управления изображением.....	37
MarqueeType.....	38
MarqueeFormat.....	38
MarqueeUnitWait	41
MarqueeRepeatWait	41
InterCharacterWait	41
BlinkInterval	41
DeviceBrightness.....	42
DisplayText (Data, Attribute) Показать Текст (Строка, Атрибут)	42
DisplayTextAT (Y, X, Data, Attribute) Показать Текст Поз (Y, X, Строка, Атрибут).....	45
RefreshWindow (Window) Обновить Окно (Окно)	48
ScrollText (Direction, Unit) Прокрутить Текст (Направление, Сдвиг)	48

ClearText () ОчиститьТекст ().....	48
Clear () Очистить ().....	49
Прочие методы	49
SelfTest () СамоТестирование ()	49
OpenDrawer () ОткрытьЯщик ()	49
Подключение драйвера	51
1С: Предприятие v.7.7	51
1С: Предприятие v.8.x	51
Microsoft VBA (Excel, Word и др.)	53
Borland Delphi, C++ Builder.....	54
Приложение 1. Коды и описание ошибок	55

Введение

Сокращения

ККМ	Контрольно-кассовая машина
ФР	Фискальный регистратор
ПК	Персональный компьютер
ПО	Программное обеспечение
ВК	Внешняя компонента для системы программ «1С: Предприятие» 7.7, 8.x
ЛУ	Логическое устройство
ДЯ	Денежный ящик
Дисплей	Дисплей покупателя
Оборудование	Денежный ящик и/или дисплей покупателя
ОС	Операционная система
Win32	Семейство ОС Windows 2000 / 2003 x86 / XP x86 / Vista x86 / 7 x86 / 7 x64 / 8 x86 / 8 x64

Условные обозначения



Информация, выделенная таким образом, является важной и требует обязательного прочтения и/или выполнения.



Информация, отмеченная такой иконкой, носит ознакомительный и/или рекомендательный характер.



Информация, отмеченная такой иконкой, является примером использования настройки или механизма работы.

О руководстве

Данное руководство программиста является описанием продукта «АТОЛ: Драйвер дисплеев покупателя» v. 8.0. При написании руководства подразумевалось, что читатель имеет навыки программирования на одном или нескольких языках программирования для операционных систем: Windows 2000 / 2003 x86 / XP x86 / Vista x86 / 7 x86 / 7 x64 / 8 x86 / 8 x64, а также знаком с используемым оборудованием.

Ввиду универсальности драйверов не все функциональные возможности оборудования могут быть реализованы в драйвере. Компания АТОЛ всегда стремится к поддержке всех функциональных возможностей конкретной модели оборудования, но оставляет за собой право реализации тех функций, которые считает необходимыми. Все возможности драйвера подробно изложены в данном документе, свободно доступном на сайте компании АТОЛ, с которым можно ознакомиться до приобретения драйвера.

Что такое «Драйвер дисплеев покупателя»?

«АТОЛ: Драйвер дисплеев покупателя» – программная компонента (драйвер), предназначенная для работы с дисплеями покупателя и денежными ящиками.

Дисплей покупателя – алфавитно-цифровое табло с различными режимами вывода текста и набором индикаторов, имеющих фиксированные значения. В зависимости от модели, на дисплее можно регулировать яркость свечения, выводить мигающие символы, выводить текст в режимах телетайпа и бегущей строки. Дисплеи покупателя могут быть использованы для демонстрации различной информации в режиме бегущей строки или же, в сочетании с POS-терминалами и кассами, для вывода результатов кассовых операций.

Денежный ящик – механическое устройство, предназначенное для хранения наличных денег в течение рабочего дня. Денежные ящики обычно имеют запирающий механический замок и электрическое реле. Первый позволяет блокировать доступ к содержимому в моменты времени, когда кассир отлучается надолго от своего рабочего места. Реле позволяет открывать ящик сигналами от ПК или ККМ.

Драйвер представляет собой внешнюю компоненту для системы «1С: Предприятие» и сервер OLE Automation и ActiveX, работающий под управлением ОС Windows 2000 / 2003 x86 / XP x86 / Vista x86 / 7 x86 / 7 x64 / 8 x86 / 8 x64. Он может использоваться в любых средах разработки, поддерживающих технологию OLE Automation:

- 1С: Предприятие;
- Borland Delphi;
- Borland C++ Builder;
- Microsoft Visual C++;
- Microsoft Visual Fox Pro;
- Microsoft Visual Basic;
- Приложения Microsoft Office с VBA (Excel, Word, Access и др.);
- Navision Axapta и др.

Более подробную информацию о драйверах торгового оборудования и их взаимодействии с оборудованием и программным обеспечением можно найти в документе «АТОЛ: Драйвер торгового оборудования».

Интерфейс драйвера

Интерфейс драйвера состоит из методов и свойств. Все методы представляют собой функции без параметров, возвращающие результат выполнения операции.

Описание метода выглядит следующим образом:

```
// Язык Pascal
function Имя( ) : integer;

// Язык C
int function Имя( );

// Язык Basic
Function Имя( ) As Integer
```

Для большинства методов требуется указывать параметры. Например, для метода печати строки это строка, которая должна быть напечатана, межстрочный интервал, шрифт и т.д.

В драйвере для передачи подобных данных используются свойства. Фактически, это глобальные переменные драйвера, в которые можно записать или считать значение. Однажды установленное свойство сохраняет свое значение до последующего изменения или выгрузки самого драйвера.

Рассмотрим функцию «Добавить логическое устройство» (AddDevice ()), которой необходимо передать название устройства, а результатом работы будет индекс созданного логического устройства.

Функция могла бы выглядеть следующим образом:

```
// Язык Pascal
function Print (ASrt: string) : integer;

// Язык C
int Print (char * AStr);

// Язык Basic
Function Print (AStr As string) As integer;
```

Но в драйвере используются два свойства CurrentDeviceName и CurrentDeviceIndex. Таким образом, вызов этого метода будет выглядеть следующим образом:

```
Если Драйвер.AddDevice() <> 0 Тогда
    // Сообщение пользователю об ошибке
    // Выход из программы
КонецЕсли;

Драйвер.CurrentDeviceName = "Устройство на COM1";

Сообщение ("Создано логическое устройство с индексом №" +
ЧислоВСтроку (Драйвер.CurrentDeviceIndex);
```

Данный подход позволяет:

- устанавливать значения параметров (свойств) перед выполнением метода в любом порядке;
- обращаться к параметрам (свойствам) по именам;
- устанавливать значения только необходимым параметрам (свойствам), а в остальных оставлять последние значения.

Как уже было сказано выше, все методы драйвера являются функциями, возвращающими код результата. Для удобства код результата также заносится в свойство `ResultCode`, а его описание – в `ResultDescription`.

Структура описания методов и свойств

При описании методов и свойств будет использоваться следующая структура.

MethodName (Param1,.., ParamN)
НазваниеМетода (Парам1,.., ПарамN)

Подробное описание использования метода.

Название	Тип	Дост.	Значения
Параметры			
Param1 Парам1	Int	RW	Значения параметра: 0 ... 100
Вызов некоторых методов должен производиться с указанными через запятую в скобках параметрами. Например: <pre>DisplayText (Data, Attribute);</pre> Если этого раздела таблице нет, то метод вызывается без параметров			
Входные свойства			
InputProperty ВходноеСвойство	Int	RW	Значения свойства: 0 ... 100
Свойства, значения которым необходимо присвоить до вызова метода. Если этого раздела в таблице нет, то для вызова данного метода не требуется изменение значения никаких свойств			
Выходные свойства			
OutputProperty ВыходноеСвойство	Log	R	Значения свойства: TRUE / FALSE

Название	Тип	Дост.	Значения
Свойства, принимающие значения после вызова метода. Если этого раздела в таблице нет, то данный метод не изменяет значения никаких свойств			

Описание параметров

Подробное описание параметров вызова метода.

Param1

Парам1

Подробное описание параметра вызова метода.

Описание свойств

Подробное описание входных и выходных свойств.

InputProperty

ВходноеСвойство

[ВХ]

Подробное описание использования входного свойства.

OutputProperty

ВыходноеСвойство

[ВЫХ]

Подробное описание использования выходного свойства.

Возможные ошибки

Описание характерных ошибок для данного метода.

Поддерживаемое оборудование

Модели дисплеев покупателя и денежных ящиков, поддерживающие данный метод. Если данного раздела нет, то метод поддерживается всеми моделями дисплеев покупателя.



Пример работы, включает следующие элементы:

```

InputProperty = "Значение";           // Поясняющие комментарии к разделу
Methodname (Param1, Param2);         // Комментарии к свойству
Methodname (Param1, Param2);         // Комментарии к методу

```

Классификация свойств

По типу:

Тип	Описание
Int	Целое / Integer Целое 32-битное число со знаком. Диапазон значений: -2147483648 ... 2147483647
Dbl	Дробное / Double Дробное 64-битное число со знаком. Диапазон значений: $5,0 \times 10^{-324} \dots 1,7 \times 10^{+308}$, точность 15 ... 16 знаков после дес. запятой. Для дробных величин допускается погрешность не более 0,0001. Значения с недопустимой погрешностью рассматриваются как «недопустимое значение» (ошибка - 6). Допустимая точность описана для каждого из свойств отдельно
Str	Строка / String Строка символов
Log	Логическое / Logical Целое число, интерпретируемое как «ЛОЖЬ (FALSE)» при значении 0 и «ИСТИНА (TRUE)» в остальных случаях

По доступу:

Дост.	Описание
R	Только для чтения
RW	Для чтения и записи
W	Только для записи

Отличия от драйвера версии 6

Драйвер версии 8 создавался так, чтобы обеспечить максимальную совместимость с драйвером версии 6, однако были некоторые нюансы, которые потребовали значительного изменения логики работы. В результате драйвер версии 8 позволяет создавать два типа объектов драйвера:

- Драйвер дисплеев покупателя v.8 (совместимый с v.6).
- Драйвер дисплеев покупателя v.8.

Первый обеспечивает максимальную совместимость с драйвером 6-й версии, второй – имеет некоторые отличия:

Функция	Поведение драйвера, совместимого с v.6	Поведение драйвера, не совместимого с v.6
Загрузка параметров ЛУ	При создании объекта драйвера автоматически происходит загрузка параметров из реестра (если приложение не внесено в список исключений)	При создании драйвера не происходит загрузка параметров. КПО должно самостоятельно вызвать метод загрузки параметров из реестра, либо передать строку настроек, либо создать ЛУ и передать настройки через соответствующие свойства драйвера
Создание нескольких экземпляров драйвера в одном процессе	Все экземпляры драйвера работают с единственным «внутренним» объектом. В результате невозможно осуществлять параллельную (многопоточную) работу с различными устройствами в одном процессе	Все экземпляры драйвера независимы. Возможна параллельная (многопоточная) работа нескольких экземпляров драйвера с различными устройствами

Состав методов и свойств для обоих типов объектов драйвера одинаков.



В системе 1С Предприятие v.7.7 не рекомендуется создавать объект драйвера, не совместимый с v.6. Это обусловлено особенностью данной версии 1С, приводящей к созданию нескольких экземпляров драйвера.

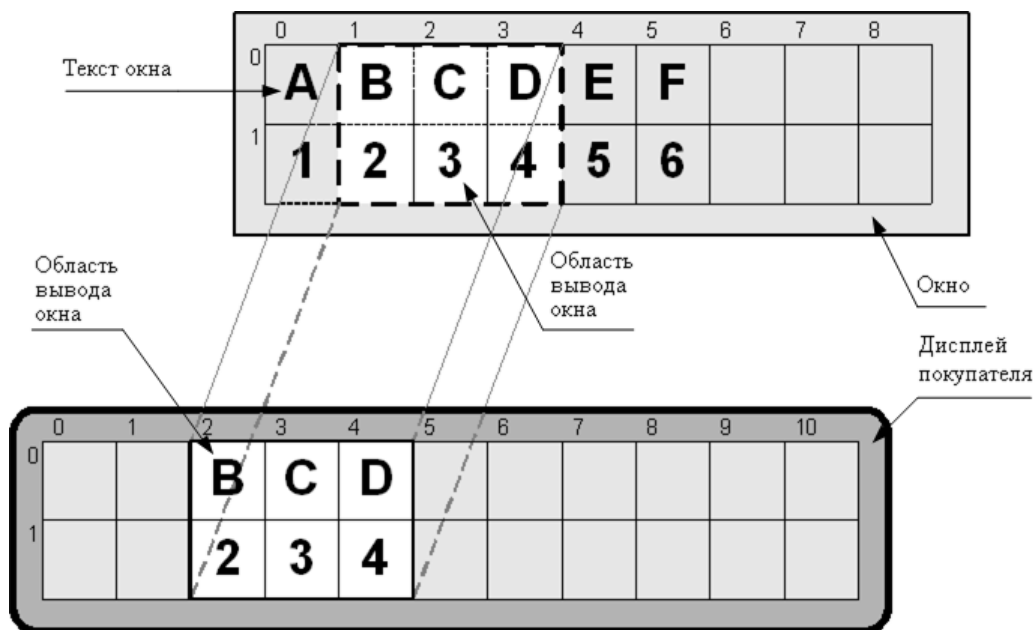
Определения

Перед тем как перейти непосредственно к описанию интерфейса работы драйвера, необходимо дать некоторые определения для понимания работы драйвера.

Окно. Набор текстовых строк одинаковой длины (не менее одной строки и не менее одного символа в строке). Размеры окна могут быть меньше, равняться или превосходить размеры дисплея.

Область вывода окна. Прямоугольная часть окна, которая будет выводиться на дисплей.

Если рассматривать дисплей как трафарет, то область вывода – прямоугольный вырез, через который видно содержимое подложки-окна:



Общие свойства

Подробное описание входных и выходных свойств приводится в каждом методе. В целях снижения громоздкости и уменьшения количества повторяющегося текста, описание некоторых свойств приведено один раз – в данном разделе.

Для этих свойств справедливо следующее:

1. Назначение этих свойств не зависит от использующего их метода.
2. В большинстве приложений нет необходимости менять эти свойства постоянно при вызове каждого метода, как правило, их значения выставляются один раз - сразу после загрузки драйвера.

Название	Тип	Дост.	Значения
ResultCode Результат	Int	R	Код ошибки. См. «Приложение 1. Коды и описание ошибок»
ResultDescription ОписаниеРезультата	Str	R	Описание кода ошибки. См. «Приложение 1. Коды и описание ошибок»

Описание свойств

ResultCode

Результат

Свойство содержит код результата выполнения последней операции (вызова метода, записи или чтения свойства драйвера). Если ошибки не произошло, то значение данного свойства устанавливается в 0 («Ошибок нет»).

Выполнение метода. Если значение свойства, используемого методом, не подходит для данного метода вообще (для случаев, когда разные методы используют разные диапазоны свойств), выставляются `ResultCode = - 6` («Недопустимое значение»). Если значение свойства подходит методу в принципе, происходит определение типа устройства. После этого, если значение не подходит данной модели, выставляется `ResultCode = -12` («Не поддерживается в данной версии оборудования»).

Запись свойства. Если записываемое значение свойства не попадает в диапазон допустимых значений ни для одной из моделей, поддерживаемых драйвером, оно не запоминается и в `ResultCode` записывается «-6».

ResultDescription

ОписаниеРезультата

Свойство содержит строку с описанием на русском языке ошибки, возникшей в результате последней операции. Может в готовом виде использоваться для выдачи пользователю предупреждающих сообщений.

Системные свойства

В данном разделе описана группа свойств, содержащих информацию о драйвере.

Название	Тип	Дост.	Значения
Version Версия	Str	R	Версия драйвера
ServerVersion ВерсияСервера	Str	R	Версия сервера
ApplicationHandle	Int	RW	Дескриптор главного окна клиентского приложения
DeviceDescription ОписаниеУстройства	Str	R	Название драйвера
IsDemo ДемонстрационныйРежим	Log	R	Флаг работы в демонстрационном режиме

Описание свойств

Version

Версия

Свойство содержит версию данного драйвера.

ServerVersion

ВерсияСервера

Свойство содержит версию используемого сервера оборудования. Если сервер не был найден и/или подключен, то свойство принимает значение «пустая строка».

Очевидно, что если два ЛУ настроены на различные ПК (см. раздел «Логические устройства»), то при переключении текущего ЛУ содержимое свойства ServerVersion тоже будет меняться.

ApplicationHandle

После загрузки драйвера в это свойство можно записать дескриптор главного окна приложения-клиента. Это предотвратит появление отдельных кнопок в панели задач при отображении визуальной страницы свойств и других окон драйвера.

При использовании драйвера в качестве внешней компоненты данное свойство не поддерживается, так как драйвер самостоятельно при загрузке инициализирует свойство корректным значением.

При записи значения в данное свойство следует проявлять особую аккуратность, так как запись некорректного значения может привести к нарушениям работы системы.

DeviceDescription

ОписаниеУстройства

Название драйвера: «Драйвер дисплеев покупателя».

IsDemo

ДемонстрационныйРежим

Если свойство IsDemo = TRUE, то драйвер не обнаружил электронного ключа защиты и работает в демонстрационном режиме.

Информационные свойства модели

В данном разделе описана группа свойств, содержащих информацию о подключенном к ПК оборудовании.

Свойства, которые позволяют клиентскому ПО определить некоторые возможности используемого драйвера и/или аппаратуры автоматически, при изменении значения свойства Model. Наличие таких свойств позволяет создавать более корректный аппаратно независимый код клиента.

Название	Тип	Дост.	Значения
CapBlink ЕстьМигание	Int	R	Мигание символов:

Название	Тип	Дост.	Значения
			0 – не поддерживается; 1 – все символы; 2 – каждый символ
CapBrightness ЕстьЯркость	Log	R	Регулирование яркости символов: FALSE / TRUE
CapCharacterSet НаборСимволовПоУмолчанию	Int	R	Набор символов по умолчанию
CharacterSetList СписокНаборовСимволов	Str	R	Коды поддерживаемых наборов символов
CapDescriptors ЕстьДескрипторы	Log	R	Работа с дескрипторами: FALSE / TRUE
DeviceDescriptors КолвоДескрипторов	Int	R	Количество дескрипторов дисплея
CapHMarquee ЕстьГоризБегСтрока	Log	R	Бегущая строка с горизонтальной прокруткой: FALSE / TRUE
CapVMarquee ЕстьВертикБегСтрока	Log	R	Бегущая строка с вертикальной прокруткой: FALSE / TRUE
CapICharWait ЕстьЗадержкаМеждуСимволами	Log	R	Вывод символов с задержкой: FALSE / TRUE
CapDownloadFonts ЕстьЗагрузкаШрифтов	Log	R	Загрузка шрифтов: FALSE / TRUE
CapSelfTest ЕстьСамотестирование	Log	R	Режим самотестирования: FALSE / TRUE
DeviceWindows КолвоОкон	Int	R	Количество окон для текущего ЛУ
DeviceColumns КолвоСтолбцовДисплея	Int	R	Ширина дисплея в символах
DeviceRows КолвоСтрокДисплея	Int	R	Высота дисплея в символах
GetDeviceColumns(Row) КолвоСтолбцовВСтроке(Строка)	Int	R	Максимальное количество символов в строке

Название	Тип	Дост.	Значения
CapStatus ЕстьСтатус	Log	R	Определять состояние ящика: FALSE / TRUE

Описание свойств

CapBlink

ЕстьМигание

- Если CapBlink = 0, то дисплей не поддерживает мигание символов.
- Если CapBlink = 1, то дисплей может выводить мигающими только все символы сразу.
- Если CapBlink = 2, то каждый символ может мигать отдельно.

Для данного драйвера: CapBlink = 1.

CapBrightness

ЕстьЯркость

- Если CapBrightness = TRUE, то данный дисплей позволяет регулировать яркость символов.
- Если CapBrightness = FALSE, то данный дисплей не позволяет регулировать яркость символов.

Для данного драйвера: CapBrightness = TRUE.

CapCharacterSet

НаборСимволовПоУмолчанию

Свойство содержит описание набора символов по умолчанию. Свойство может принимать следующие значения:

Значение	Назначение
0	Набор символов по умолчанию содержит цифры, пробел, «-» и «.» Данное значение возвращается для моделей дисплеев покупателя «Меркурий ДП-02» и «Меркурий ДП-03»
1	Набор символов по умолчанию содержит заглавные буквы, цифры, пробел, минус и точку. Данное значение возвращается для модели дисплея покупателя «Меркурий ДП-01»

Значение	Назначение
998	Набор символов по умолчанию содержит все символы с ASCII – кодами от 20H до 7FH. Данное значение возвращается для моделей дисплеев покупателя «DPD-201», «VF-2029», «VF-2021» и «Flytech»
10	Набор символов по умолчанию содержит все символы с ASCII – кодами от 20H до 7FH и символы японского письма с кодами A1H – DFH
11	Набор символов по умолчанию содержит все символы с ASCII – кодами от 20H до 7FH и кодовую страницу 932 (японское письмо)

Набор символов по умолчанию может содержать символы дополнительно к перечисленным выше.

CharacterSetList

СписокНаборовСимволов

Свойство содержит строку, в которой перечислены поддерживаемые номера кодовых таблиц символов.

Модель	Номера кодовых таблиц символов
Datcs DPD-201	437, 866
EPSON протокол (Firich, POSIFLEX, NCR 7446, NCR 7443 и др. модели, работающие по EPSON протоколу)	437, 850, 852, 860, 863, 865, 866, 932
Меркурий ДП-01	866
Меркурий ДП-02	866
Меркурий ДП-03	866
Flytech	866
GIGATEK DSP800/850	865, 866
Штрих-FrontMaster	437, 850, 852, 860, 863, 865, 866, 932
EPSON протокол USA (POSIFLEX-PD302C и др. модели, работающие по EPSON USA протоколу)	437
IPC	866
GIGATEK DSP820	865, 866
TEC LIUST51	866

Модель	Номера кодовых таблиц символов
OMRON DP75-21	866
NCR 597X	866
Штрих-miniPOSII PRO	437
Posiflex PD-201/PD-309/PD-320	437, 850, 852, 860, 863, 865, 866, 932
Cipher	437, 850, 852, 860, 863, 865, 866, 932
Posiflex PD-2800	437, 850, 852, 860, 863, 865, 866, 932

CapDescriptors

ЕстьДескрипторы

- Если CapDescriptors = TRUE, то данный дисплей поддерживает работу с дескрипторами.
- Если CapDescriptors = FALSE, то данный дисплей не поддерживает работу с дескрипторами.

Дескрипторы – небольшие информационные поля дисплея, отражающие режим, подобно индикатору «М» на калькуляторе, сигнализирующему о наличии данных в памяти.

Для данного драйвера: CapDescriptors = FALSE.

DeviceDescriptors

КолвоДескрипторов

Свойство DeviceDescriptors содержит количество дескрипторов дисплея.

Для данного драйвера: DeviceDescriptors = 0.

CapHMarquee

ЕстьГоризБегСтрока

- Если CapHMarquee = TRUE, то данный дисплей поддерживает бегущую строку с горизонтальной прокруткой.
- Если CapHMarquee = FALSE, то данный дисплей не поддерживает бегущую строку с горизонтальной прокруткой.

Для данного драйвера: CapHMarquee = TRUE.

CapVMarquee

ЕстьВертикБегСтрока

- Если CapVMarquee = TRUE, то данный дисплей поддерживает бегущую строку с вертикальной прокруткой.
- Если CapVMarquee = FALSE, то данный дисплей не поддерживает бегущую строку с вертикальной прокруткой.

Для данного драйвера: CapVMarquee = TRUE.

CapICharWait

ЕстьЗадержкаМеждуСимволами

- Если CapICharWait = TRUE, то данный дисплей позволяет выводить символы с задержкой.
- Если CapICharWait = FALSE, то данный дисплей не позволяет выводить символы с задержкой.

Для данного драйвера: CapICharWait = FALSE.

CapDownloadFonts

ЕстьЗагрузкаШрифтов

- Если CapDownloadFonts = TRUE, то данный дисплей позволяет загружать шрифты.
- Если CapDownloadFonts = FALSE, то данный дисплей не позволяет загружать шрифты.

Firich VF-2029/2021: CapDownloadFonts = TRUE

Для остальных дисплеев: CapDownloadFonts = FALSE.

CapSelfTest

ЕстьСамотестирование

- Если CapSelfTest = TRUE, то данный дисплей поддерживает режим самотестирования.
- Если CapSelfTest = FALSE, то данный дисплей не поддерживает режим самотестирования.

Для данного драйвера: CapSelfTest = FALSE.

Epson, EpsonUSA, Flytech, Gigatek DSP820: CapSelfTest = TRUE

Для остальных дисплеев: CapSelfTest = FALSE.

DeviceWindows

КолвоОкон

В свойстве DeviceWindows содержится количество окон, созданных для текущего логического устройства.

DeviceColumns

КолвоСтолбцовДисплея

В свойстве DeviceColumns содержится количество столбцов в дисплее (ширина дисплея).

Для данного драйвера: DeviceColumns = 20.

DeviceRows

КолвоСтрокДисплея

В свойстве DeviceRows содержится количество строк в дисплее (высота дисплея).

GetDeviceColumns(Row)

КолвоСтолбцовВСтроке (Строка)

В свойстве GetDeviceColumns (Row) содержится максимальное количество символов в строке Row.

CapStatus

ЕстьСтатус

- Если CapStatus = TRUE, то денежный ящик поддерживает проверку своего состояния.
- Если CapStatus = FALSE, то денежный ящик не поддерживает проверку своего состояния.

Для данного драйвера: CapStatus = TRUE.

Логические устройства

Логическое устройство – набор свойств драйвера, определяющих параметры связи с оборудованием. Подобных наборов (устройств) одновременно может быть от 1 до 99 штук. Это позволяет, однажды настроив несколько наборов свойств (например: номер порта ПК, скорость обмена данными с оборудованием), быстро применять необходимые параметры, просто переключая устройства.

Драйвер может хранить настройки ЛУ в системном реестре, а также передавать их через свойства DeviceSettings и DevicesSettings.

В случае использования объекта драйвера, совместимого с v.6, все данные о логических устройствах автоматически загружаются из реестра при создании экземпляра драйвера и автоматически сохраняются при разрушении. Если необходима совместимость с интерфейсом драйвера v.6, но при этом нужно, чтобы приложение, вызывающее драйвер, не обращалось к реестру автоматически, пропишите параметр вида

"Произвольное_имя_переменной"="Имя_Приложения"

в следующих разделах:

«[HKEY_CURRENT_USER\Software\ATOL\Drivers\6.0\AppNotLoadDevices]»

или

«[HKEY_LOCAL_MACHINE\Software\ATOL\Drivers\6.0\AppNotLoadDevices]»



[HKEY_CURRENT_USER\Software\ATOL\Drivers\6.0\AppNotLoadDevices]

"Frontol"="Frontol.exe"

"FrontolAdmin"="FrontolAdmin.exe"

или

[HKEY_LOCAL_MACHINE\Software\ATOL\Drivers\6.0\AppNotLoadDevices]

"Frontol"="Frontol.exe"

"FrontolAdmin"="FrontolAdmin.exe"

Имя логического устройства. Имя, которое задается пользователем и используется для удобства визуального выбора.

Номер логического устройства. Персональный номер устройства в списке существующих логических устройств драйвера. То есть, при создании нового логического устройства, ему присваивается минимальный свободный номер. При удалении логического устройства из «середины» списка номера остальных не меняются.

Индекс логического устройства. Порядковый номер устройства в списке существующих логических устройств драйвера. При создании нового логического устройства индексы пересчитываются так, чтобы номера шли по порядку. При удалении логического устройства из «середины» списка индексы изменяются таким образом, чтобы опять получился непрерывный ряд значений.

Текущее устройство. То устройство, свойства которого доступны в текущий момент для чтения и редактирования. Все методы драйвера работают со свойствами именно этого устройства. Чтобы изменить свойства другого устройства, его необходимо предварительно сделать текущим. Изменяя номер или индекс логического устройства, можно выбрать текущее устройство.



Например, существовали три ЛУ с номерами: 1, 2 и 3 с индексами 0, 1 и 2 соответственно; после удаления ЛУ с номером 2 появится «дыра», т.е. можно сделать текущим ЛУ с номером 1 или 3, но не 2. Однако индекс устройства № 3 изменился: был «2», а стал «1».

При создании нового ЛУ ему будет присвоен номер 2 и индекс 1 (у устройства №3 индекс поменяется с 1 на 2).

Логическими устройствами (добавление, удаление и т.д.) драйвер может управлять следующими способами:

1. Программно – методы AddDevice (), DeleteDevice () и т.д.
2. Визуально – метод ShowProperties ().

Последний способ является предпочтительным, так как вы избавляетесь от временных затрат на разработку собственного подобного интерфейса.

Свойства текущего ЛУ

В данном разделе описана группа свойств, используемая для изменения параметров текущего логического устройства.

Название	Тип	Дост.	Значения
CurrentDeviceIndex ИндексТекущегоУстройства	Int	RW	Индекс текущего ЛУ: 0 ... 98
CurrentDeviceNumber НомерТекущегоУстройства	Int	RW	Номер текущего ЛУ: 1 ... 99
DeviceEnabled УстройствоВключено	Log	RW	Флаг «Устройство включено»: FALSE / TRUE
MachineName НаименованиеМашины	Str	RW	Имя ПК в сети
PortNumber НомерПорта	Int	RW	Номер порта ПК: • 1001 – COM1; • ... • 1256 – COM256; • 66 – Posiflex USB; • 104 – ComProxy1; • 105 – ComProxy2
BaudRate СкоростьОбмена	Int	RW	Скорость обмена с ПК: • 4 – 2400; • 5 – 4800; • 7 – 9600; • 9 – 14400; • 10 – 19200; • 11 – 38400; • 12 – 57600; • 13 – 115200
Model Модель	Int	RW	Модель подключенного оборудования

Название	Тип	Дост.	Значения
LimitSize ОграничитьРазмер	Int	RW	Размер экрана дисплея: • 0 – 20x2; • 1 – 16x1; • 2 – 26x2
CharacterSet НаборСимволов	Int	RW	Номер кодовой таблицы символов
Parity Четность	Int	RW	Метод проверки четности: • 0 – нет (None); • 1 – нечетность (Odd); • 2 – четность (Even); • 3 – установлен (Mark); • 4 – сброшен (Space)
DataBits БитыДанных	Int	RW	Количество бит данных в байте: • 3 – 7 бит; • 4 – 8 бит
StopBits СтопБиты	Int	RW	Количество стоп-битов: • 0 – 1 бит; • 2 – 2 бита
DevicesSettings ПараметрыУстройств	Str	RW	Параметры логических устройств в виде строки
DeviceSettings ПараметрыУстройства	Str	RW	Параметры логического устройства в виде строки

Описание свойств

CurrentDeviceIndex

ИндексТекущегоУстройства

В свойство записывается индекс текущего логического устройства.

Если логическое устройство с таким индексом не может существовать, то значение свойства CurrentDeviceIndex сохраняет значение, содержавшееся до операции присвоения, а в ResultCode заносится -9.

CurrentDeviceNumber

НомерТекущегоУстройства

В свойство записывается номер текущего логического устройства.

Если логическое устройство с таким номером не может существовать, то значение свойства `CurrentDeviceNumber` сохраняет значение, содержащее до операции присвоения, а в `ResultCode` заносится -9.

DeviceEnabled

УстройствоВключено

При установлении `DeviceEnabled = TRUE` драйвер занимает порт ПК, установленный в свойстве `PortNumber`. В случае, если порт по каким-либо причинам занять не удалось, то `DeviceEnable` становится равным `FALSE`, а в `ResultCode` заносится код ошибки.

MachineName

НаименованиеМашины

В свойстве указывается сетевое имя/IP-адрес, а также IP-порт ПК, к которому подключено устройство. Если свойство пустое, то используется локальное обращение к сервисному модулю.

Если работа с устройством ведется через порт удаленной машины, то на машине-сервере (к которой подключено устройство) должен быть установлен и зарегистрирован сервисный модуль. На машине-клиенте (с которой производится обращение к устройству) должны быть установлены и зарегистрированы драйвер и сервисный модуль. На обеих машинах необходимо установить службу `DtoSvc`.

PortNumber

НомерПорта

В свойство записывается номер COM-порта ПК, к которому подключено оборудование и на работу с которым настроено данное логическое устройство.

Если `DeviceEnabled = TRUE`, то присвоение свойству `PortNumber` нового значения приводит к установке `DeviceEnabled = FALSE`, и последующей попытке восстановить `DeviceEnabled = TRUE` с новыми параметрами. После установки свойства необходимо проверить значение свойства `DeviceEnabled` для определения, удалось ли инициализировать новый порт, так как если порт занять не удалось, `DeviceEnabled` принимает `FALSE`, а `ResultCode = 0`.

BaudRate

СкоростьОбмена

В свойство записывается код скорости обмена данными с ПК.

Model

Модель

В свойство записывается значение, определяющее модель дисплея покупателя, по протоколу обмена которого должен работать драйвер с текущим логическим устройством:

Значение	Описание
0	Datecs DPD-201
1	Протокол EPSON (Firich, POSIFLEX, NCR 7446, NCR 7443 и др.)
2	Меркурий ДП-01
3	Меркурий ДП-02
4	Меркурий ДП-03
5	Flytech
6	GIGATEK DSP800/850
7	Штрих-FrontMaster
8	Протокол EPSON USA (POSIFLEX-PD302C и др.)
10	IPC
11	GIGATEK DSP820
12	TEC LIUST51
13	OMRON DP75-21
14	NCR 597X
15	Штрих-miniPOSII PRO
16	Posiflex PD-201/PD-309/PD-320
17	Cipher
18	Posiflex PD-2800

LimitSize

Ограничить Размер

Свойство содержит размер экрана дисплея покупателя.

Поддерживаемые дисплеи покупателей

Model	Название
1	EPSON (Firich, POSIFLEX, NCR 7446, NCR 7443 и др.)
8	EPSON USA (POSIFLEX-PD302C и др.)
16	Posiflex PD-201/PD-309/PD-320
17	Cipher
18	Posiflex PD-2800

CharacterSet

НаборСимволов

Свойство содержит номер кодовой таблицы символов.



Для корректного отображения информации на дисплее требуется правильно выбрать кодовую таблицу. Поддерживаемые кодовые таблицы символов смотрите в руководстве по эксплуатации вашего дисплея.

Parity

Четность

В свойство записывается значение, определяющее метод проверки четности при обмене с дисплеем.

DataBits

БитыДанных

В свойство записывается значение, определяющее параметр связи с дисплеем «Битов в данном байте».

StopBits

СтопБиты

В свойство записывается значение, определяющее параметр связи с дисплеем «Количество стоп-битов».

DevicesSettings

ПараметрыУстройств

Получение и изменение параметров логических устройств в виде строки.



```
Count=2
CurrentDeviceNumber=2
DeviceNumber0=1
DeviceName0=
PortNumber0=1
Parity0=0
BaudRate0=7
DataBits0=4
Model0=1
DownloadFonts0=False
MachineName0=
CharacterSet0=866
DeviceNumber1=2
DeviceName1=Без названия
PortNumber1=1
Parity1=0
BaudRate1=7
DataBits1=4
Model1=0
DownloadFonts1=False
MachineName1=
CharacterSet1=866
```

DeviceSettings

ПараметрыУстройства

Получение и изменение параметров логического устройства в виде строки.



Формат данных в строках параметров может изменяться в новых версиях драйвера. Поэтому не рекомендуется редактировать эти (DevicesSettings и DeviceSettings) строки в прикладном ПО.



```
DeviceNumber=2
DeviceName=Без названия
PortNumber=1
Parity=0
BaudRate=7
DataBits=4
Model=0
DownloadFonts=False
MachineName=
CharacterSet=866
```

AddDevice ()

ДобавитьУстройство ()

Метод создает новое логическое устройство и устанавливает его текущим.

Для определения числа логических устройств, существующих на данный момент, необходимо воспользоваться свойством DeviceCount. Индекс первого логического устройства равен 0, а последнего DeviceCount-1.

Название	Тип	Дост.	Значения
Выходные свойства			
CurrentDeviceIndex ИндексТекущегоУстройства	Int	RW	Индекс текущего ЛУ: 0 ... 98
CurrentDeviceNumber НомерТекущегоУстройства	Str	RW	Номер текущего ЛУ: 1 ... 99
CurrentDeviceName НаименованиеТекущегоУстройства	Str	RW	Название ЛУ
DeviceCount КоличествоУстройств	Int	R	Количество ЛУ: 1 ... 99

Описание свойств

CurrentDeviceIndex

ИндексТекущегоУстройства

[ВЫХ]

При добавлении нового логического устройства CurrentDeviceIndex увеличивается на 1.

CurrentDeviceNumber

НомерТекущегоУстройства

[ВЫХ]

Свойство CurrentDeviceNumber содержит минимальный свободный номер логического устройства.

CurrentDeviceName

НаименованиеТекущегоУстройства

[ВЫХ]

Названия логических устройств используются только для удобства пользовательского выбора.

При добавлении нового логического устройства свойство CurrentDeviceName принимает значение «Без имени».

DeviceCount

КоличествоУстройств

[ВЫХ]

При добавлении нового логического устройства DeviceCount увеличивается на единицу.

Возможные ошибки

Код	Причина
-7	Попытка создания более 99 устройств

DeleteDevice ()
УдалитьУстройство ()

Метод производит удаление текущего логического устройства.

Для определения числа логических устройств, существующих на данный момент, необходимо воспользоваться свойством DeviceCount. Индекс первого логического устройства равен 0, а последнего DeviceCount - 1.

Название	Тип	Дост.	Значения
Выходные свойства			
CurrentDeviceIndex ИндексТекущегоУстройства	Int	RW	Индекс текущего ЛУ: 0 ... 98
CurrentDeviceNumber НомерТекущегоУстройства	Str	RW	Номер текущего ЛУ: 1 ... 99
CurrentDeviceName НаименованиеТекущегоУстройства	Str	RW	Название ЛУ
DeviceCount КоличествоУстройств	Int	R	Количество ЛУ: 1 ... 99

*Описание свойств**CurrentDeviceIndex*

ИндексТекущегоУстройства

[Вых]

При удалении не последнего ЛУ, текущим становится ЛУ, следующее за удаляемым.

При удалении последнего ЛУ, текущем становится ЛУ, предшествующее удаляемому.

CurrentDeviceNumber

НомерТекущегоУстройства

[Вых]

При удалении не последнего ЛУ, текущим становится ЛУ с ближайшим наибольшим номером.

При удалении последнего ЛУ, текущем становится ЛУ предшествующее удаляемому.

DeviceCount

КоличествоУстройств

[ВЫХ]

При удалении текущего логического устройства DeviceCount уменьшается на единицу.

Возможные ошибки

Код	Причина
-8	Нельзя удалить все устройства – должно оставаться хотя бы одно

ShowProperties ()

ПоказатьСтраницуСвойств ()

Выводит на экран визуальную страницу свойств. Подробнее о странице свойств смотрите в документе «АТОЛ: Драйвер торгового оборудования».

Описание свойств

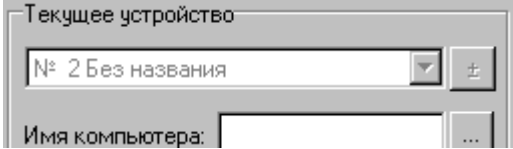
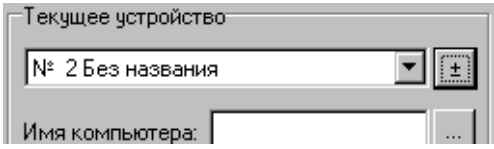
LockDevices

БлокироватьУстройства

[ВХ]

Логический параметр, определяющий возможность изменения на странице свойств параметров текущего логического устройства.

Если LockDevices = TRUE, то работа с логическими устройствами с помощью визуальной страницы свойств заблокирована, иначе (FALSE) – разрешена.

LockDevices = TRUE	LockDevices = FALSE
	

Методы работы с окнами и областями вывода

Свойства текущего окна

Название	Тип	Дост.	Значения
CurrentWindow ТекущееОкно	Int	RW	Номер текущего окна
Columns КолвоСтолбцов	Int	R	Количество столбцов в создаваемом окне (ширина окна): 0 .. Wwindow
Rows КолвоСтрок	Int	R	Количество строк в создаваемом окне (высота окна): 0 .. Hwindow
CursorColumn КурсорСтолбец	Int	RW	Номер столбца положения курсора: 0 .. Columns
CursorRow КурсорСтрока	Int	RW	Номер строки положения курсора: 0 .. Rows
CursorUpdate ОбновлятьКурсор	Log	RW	Изменять значение свойств CursorColumn и CursorRow: FALSE / TRUE

CurrentWindow

ТекущееОкно

Свойство содержит номер текущего окна вывода текста.

Для каждого окна отдельно сохраняются свойства: Columns, Rows, CursorColumn, CursorRow, CursorUpdate, MarqueeType, MarqueeFormat, MarqueeUnitWait, MarqueeRepeatWait, InterCharacterWait.

Columns

КолвоСтолбцов

Свойство содержит количество столбцов в текущем окне (ширину окна). Для окна с номером 0 (CurrentWindow = 0) всегда: Columns = DeviceColumns.

Для данного драйвера: Columns = 20.

Rows

КолвоСтрок

Свойство содержит количество строк в текущем окне (высоту окна). Для окна с номером 0 (`CurrentWindow = 0`) всегда: `Rows = DeviceRows`.

Для данного драйвера: `Rows = 2`.

CursorColumn

КурсорСтолбец

Свойство задает номер столбца (абсциссу) позиции текущего окна, в которую будет выведен следующий символ. `CursorColumn` всегда должен находиться в диапазоне от 0 до `Columns`.

Свойство `CursorColumn` является параметром текущего окна, то есть при переключении окон (изменении содержимого свойства `CurrentWindow` или удалении текущего окна) соответственно изменяется и содержимое этого свойства. `CursorColumn` также изменяется методами `DisplayTextAt()`, `ClearText()`.

CursorRow

КурсорСтрока

Свойство содержит номер строки (ординату) позиции текущего окна, в которую будет выведен следующий символ. `CursorRow` всегда должен находиться в диапазоне от 0 до `Rows`.

Свойство `CursorRow` является параметром текущего окна, то есть при переключении окон (изменении содержимого свойства `CurrentWindow` или удалении текущего окна) соответственно изменяется и содержимое этого свойства. `CursorRow` также изменяется методами `DisplayTextAt()`, `ClearText()`.

CursorUpdate

ОбновлятьКурсор

- Если `CursorUpdate = TRUE`, то методы `DisplayText (Data, Attribute)` и `DisplayTextAT (Y, X, Data, Attribute)` изменяют значения свойств `CursorColumn` и `CursorRow`, устанавливая их на позицию за последним выведенным символом.
- Если `CursorUpdate = FALSE`, то методы `DisplayText (Data, Attribute)` и `DisplayTextAT (Y, X, Data, Attribute)` значения свойств `CursorColumn` и `CursorRow` оставляют без изменений.

Свойство `CursorUpdate` является параметром текущего окна, то есть при переключении окон (изменении содержимого свойства `CurrentWindow` или удалении текущего окна) соответственно изменяется и содержимое этого свойства.

CreateWindow (Yview, Xview, Hview, Wview, Hwindow, Wwindow) СоздатьОкно (Yобласти, Xобласти, Vобласти, Шобласти, Вокна, Шокна)

Создать новое окно и соответствующую ему область вывода на дисплее.

Высота и ширина создаваемого окна задаются параметрами Hwindow и Wwindow соответственно (координаты внутри окна принадлежат диапазонам: от 0 до Hwindow - 1 для строк и от 0 до Wwindow - 1 для столбцов). Высота и ширина области вывода окна задаются параметрами Hview и Wview соответственно. Координаты левого верхнего угла области вывода задаются параметрами Xview (столбец) и Yview (строка).

Новое окно становится текущим.

Для каждого создаваемого окна имеется собственный набор из 10 выходных свойств.

Название	Тип	Дост.	Значения
Параметры			
Yview Yобласти	Int	W	Координата расположения левого верхнего угла области вывода по вертикали (строка): 0...Wwindow - 1
Xview Xобласти	Int	W	Координата расположения левого верхнего угла области вывода по горизонтали (столбец): 0...Hwindow - 1
Hview Vобласти	Int	W	Высота области вывода создаваемого окна
Wview Шобласти	Int	W	Ширина области вывода создаваемого окна
Hwindow Вокна	Int	W	Высота создаваемого окна
Wwindow Шокна	Int	W	Ширина создаваемого окна
Выходные свойства			
Columns КолвоСтолбцов	Int	R	Количество столбцов в созданном окне (ширина окна): Wwindow
Rows КолвоСтрок	Int	R	Количество строк в созданном окне (высота окна): Hwindow

Название	Тип	Дост.	Значения
CursorColumn КурсорСтолбец	Int	RW	Номер столбца положения курсора: 0
CursorRow КурсорСтрока	Int	RW	Номер строки положения курсора: 0
CursorUpdate ОбновлятьКурсор	Log	RW	Изменять значение свойств CursorColumn и CursorRow: TRUE
MarqueeType ТипБегСтроки	Int	RW	Тип бегущей строки: 0
MarqueeFormat ФорматБегСтроки	Int	RW	Формат бегущей строки: 1
MarqueeUnitWait ЗадержкаПоказаБегСтроки	Int	RW	Задержка в миллисекундах между последовательными сдвигами столбцов и строк, (мс): 0
MarqueeRepeatWait ЗадержкаПовтораБегСтроки	Int	RW	Задержка в миллисекундах между сдвигом последнего символа или строки окна в область вывода окна и повторением запуска бегущей строки с начала, (мс): 0
InterCharacterWait ЗадержкаМеждуСимволами	Int	RW	Задержка между выводом символов для создания эффекта телетайпа, (мс): 0

DestroyWindow () ***УдалитьОкно ()***

По этой команде драйвер уничтожает текущее окно (см. свойство CreateWindow (Yview, Xview, Hview, Wview, Hwindow, Wwindow)). При этом отображаемая на дисплее информация не изменяется, текущим становится окно с номером 0, а свойства, характерные для окна, принимают значения нулевого окна.

Название	Тип	Дост.	Значения
Входные свойства			
CurrentWindow ТекущееОкно	Int	RW	Номер текущего окна

Описание свойств

CurrentWindow

ТекущееОкно

Свойство содержит номер текущего окна вывода текста.

AddViewPort (Yview, Xview, Hview, Wview)

ДобавитьОбластьВывода (Yобласти, Xобласти, Вобласти, Шобласти)

С помощью этой команды можно добавить область вывода, в которой будет отображено сообщение. Высота и ширина области вывода задаются параметрами Hview, Wview соответственно. Координаты левого верхнего угла области вывода задаются параметрами Xview (столбец) и Yview (строка).

Название	Тип	Дост.	Значения
Параметры			
Yview Yобласти	Int	RW	Координата расположения левого верхнего угла области вывода по вертикали (строка): $0 \dots Wwindow - 1$
Xview Xобласти	Int	RW	Координата расположения левого верхнего угла области вывода по горизонтали (столбец): $0 \dots Hwindow - 1$
Hview Вобласти	Int	RW	Высота области вывода создаваемого окна
Wview Шобласти	Int	RW	Ширина области вывода создаваемого окна



При добавлении пересекающихся областей вывода общая область вывода будет равна их объединению.

ClearViewPorts ()

ОчиститьОбластьВывода ()

С помощью этой команды можно удалить все области вывода, связанные с текущим окном.

Название	Тип	Дост.	Значения
Входные свойства			
CurrentWindow ТекущееОкно	Int	RW	Номер текущего окна

Описание свойств

CurrentWindow

ТекущееОкно

[BX]

Свойство содержит номер текущего окна вывода текста.

Методы управления изображением

Свойства изображения

Название	Тип	Дост.	Значения
MarqueeType ТипБегСтроки	Int	RW	Тип бегущей строки: • 0 – строка не запущена; • 1 – сдвиг строки вверх; • 2 – сдвиг строки вниз; • 3 – сдвиг строки влево; • 4 – сдвиг строки вправо
MarqueeFormat ФорматБегСтроки	Int	RW	Формат бегущей строки: • 1 – текст появляется по одному символу, затем перемещается к началу; • 2 – текст появляется сразу и затем удаляется с экрана
MarqueeUnitWait ЗадержкаПоказаБегСтроки	Int	RW	Задержка между последовательными сдвигами столбцов и строк, (мс)
MarqueeRepeatWait ЗадержкаПовтораБегСтроки	Int	RW	Задержка между сдвигом последнего символа или строки окна в область вывода окна и повторением запуска бегущей строки с начала, (мс)
InterCharacterWait ЗадержкаМеждуСимволами	Int	RW	Задержка между выводом символов для создания эффекта телетайпа, (мс)

Название	Тип	Дост.	Значения
BlinkInterval ИнтервалМигания	Int	RW	Период мигания символов (мс)
DeviceBrightness Яркость	Int	RW	Уровень яркости дисплея (%)

MarqueeType

ТипБегСтроки

Свойство задает тип бегущей строки для текущего окна:

Значение	Описание
0	Бегущая строка не запущена
1	Сдвиг бегущей строки вверх
2	Сдвиг бегущей строки вниз
3	Сдвиг бегущей строки влево
4	Сдвиг бегущей строки вправо

MarqueeFormat

ФорматБегСтроки

Свойство задает формат бегущей строки для текущего окна и может принимать следующие значения:

Значение	Описание
1	Текст появляется по одному символу, а затем перемещается к началу
2	Текст появляется сразу, а затем удаляется с экрана

MarqueeFormat используется только в режиме включенной бегущей строки (MarqueeType <> 0).

// Пример1.

// Дисплей имеет две строки (0 и 1) по 20 столбцов (0..19)

// Создаем окно 2x5, область вывода 2x3, коор.0x2

CreateWindow (0, 2, 2, 3, 2, 5);

DisplayText ("0123456789", 0); // Поместим в окно текст

Окно:

Дисплей:

	0	1	2	3	4
0	0	1	2	3	4
1	5	6	7	8	9

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			0	1	2															
1			5	6	7															

DisplayTextAT (0, 4, «AB», 0); // Поместить в окно текст «AB»

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			0	1	2															
1			B	6	7															

MarqueeType = 3; //сдвиг бегущей строки влево
 MarqueeFormat = 1; // формат бегущей строки
 MarqueeRepeatWait = 500; // задержка повтора 500 мс
 MarqueeUnitWait = 500; // задержка показа 500 мс
 DisplayTextAT (0, 4, «AB», 0); //поместить в окно текст
 // По истечении времени задержки (MarqueeUnitWait = 500)

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0					0															
1					B															

// По истечении времени задержки (MarqueeUnitWait = 500)

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0				0	1															
1				B	6															

// По истечении времени задержки (MarqueeUnitWait = 500)

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			0	1	2															
1			B	6	7															

// По истечении времени задержки (MarqueeUnitWait = 500)

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			1	2	3															
1			6	7	8															

// По истечении времени задержки (MarqueeUnitWait = 500)

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			2	3	A															
1			7	8	9															

// По истечении времени задержки (MarqueeUnitWait = 500)

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0					0															
1					B															

// Пример2.

// Дисплей имеет две строки (0 и 1) по 20 столбцов (0..19)

// Создаем окно 2x5, область вывода 2x3, коор.0x2

CreateWindiw (0, 2, 2, 3, 2, 5);

DisplayText ("0123456789", 0); // Поместим в окно текст

Окно:

	0	1	2	3	4
0	0	1	2	3	4
1	5	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			0	1	2															
1			5	6	7															

DisplayTextAT (0, 4, «AB», 0); // Поместить в окно текст «AB»

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			0	1	2															
1			B	6	7															

MarqueeType = 3; // сдвиг бегущей строки влево

MarqueeFormat = 2; // формат бегущей строки

MarqueeRepeatWait = 500; // задержка повтора 500 мс

MarqueeUnitWait = 500; // задержка показа 500 мс

DisplayTextAT (0, 4, «AB», 0); // поместить в окно текст

// По истечении времени задержки (MarqueeUnitWait = 500)

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			0	1	2															
1			B	6	7															

// По истечении времени задержки (MarqueeUnitWait = 500)

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			1	2	3															
1			6	7	8															

// По истечении времени задержки (MarqueeUnitWait = 500)

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			2	3	A															
1			7	8	9															

// По истечении времени задержки (MarqueeUnitWait = 500)

Окно:

	0	1	2	3	4
0	0	1	2	3	A
1	B	6	7	8	9

Дисплей:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0			0	1	2															
1			B	6	7															

MarqueeUnitWait

ЗадержкаПоказаБегСтроки

Свойство задает задержку (в миллисекундах) между последовательными сдвигами столбцов или строк.

MarqueeRepeatWait

ЗадержкаПовтораБегСтроки

Свойство задает задержку (в миллисекундах) между сдвигом последнего символа или строки окна в область вывода окна и повторением запуска бегущей строки с начала.

InterCharacterWait

ЗадержкаМеждуСимволами

Свойство задает задержку (в миллисекундах) между выводом символов для создания эффекта телетайпа. Использование `InterCharacterWait` допустимо, только если не используется режим бегущей строки (`MarqueeType = 0`).

Свойство `InterCharacterWait` является параметром текущего окна, то есть при переключении окон (изменении содержимого свойства `CurrentWindow` или удалении текущего окна) соответственно изменяется и содержимое этого свойства.

Если `CursorUpdate = TRUE`, то вывод текста изменяет содержимое свойств `CursorColumn` и `CursorRow` перед возвратом из метода (даже если вывод текста не закончен).

Данное свойство зарезервировано для будущих версий.

BlinkInterval

ИнтервалМигания

Свойство задает период (в миллисекундах) мигания символов. Возможны следующие случаи использования свойства:

- Если `CapBlink = 0`, то мигание не поддерживается.
- Если `CapBlink = 1`, то поддерживается мигание только всех символов.
- Если `CapBlink = 2`, то поддерживается мигание отдельных символов.

Datecs DPD-201: не используется.

DeviceBrightness

Яркость

Свойство содержит уровень яркости дисплея в процентах.

Для любого дисплея допустимы хотя бы два значения:

- DeviceBrightness = 0 (0% – символ не виден).
- DeviceBrightness = 100 (100% – максимальная яркость, с которой дисплей может выводить информацию).

Если свойство CapBrightness = TRUE, то дисплей позволяет регулировать яркость символов (устанавливать в DeviceBrightness дополнительные значения из диапазона от 0 до 99).

Если дисплей может поддерживать не 100, а меньше градаций яркости – в этом случае устанавливается ближайшая возможная яркость.

При DeviceBrightness = 0 производится очистка экрана.

DisplayText (Data, Attribute)
ПоказатьТекст (Строка, Атрибут)

По этой команде драйвер помещает в окно строку, задаваемую параметром Data, и с типом мерцания из Attribute.

Текст выводится, начиная с позиции (CursorColumn, CursorRow), и продолжается в соседних столбцах. Когда достигнут последний столбец текущей строки, вывод текста переходит на новую строку. Если вывод достиг конца окна, то окно сдвигается вверх на одну строку и нижняя строка заполняется пробелами.

Название	Тип	Дост.	Значения
Параметры			
Data Строка	Str	RW	Строка, выводимая в окно
Attribute Атрибут	Int	RW	Тип мерцания: • 0 – мерцания нет; • 1 – аппаратное мерцание; • 2 – мерцание текста; • 3 – мерцание окна
Входные свойства			
CurrentWindow ТекущееОкно	Int	RW	Номер текущего окна

Название	Тип	Дост.	Значения
CursorColumn КурсорСтолбец	Int	RW	Номер столбца положения курсора: 0 .. Columns
CursorRow КурсорСтрока	Int	RW	Номер строки положения курсора: 0 .. Rows
CursorUpdate ОбновлятьКурсор	Log	RW	Изменять значение свойств CursorColumn и CursorRow: FALSE / TRUE
MarqueeType ТипБегСтроки	Int	RW	Тип бегущей строки: • 0 – строка не запущена; • 1 – сдвиг строки вверх; • 2 – сдвиг строки вниз; • 3 – сдвиг строки влево; • 4 – сдвиг строки вправо
MarqueeFormat ФорматБегСтроки	Int	RW	Формат бегущей строки: • 1 – текст появляется по одному символу, затем перемещается к началу; • 2 – текст появляется сразу и затем удаляется с экрана
MarqueeUnitWait ЗадержкаПоказаБегСтроки	Int	RW	Задержка между последовательными сдвигами столбцов и строк, (мс)
MarqueeRepeatWait ЗадержкаПовтораБегСтроки	Int	RW	Задержка между сдвигом последнего символа или строки окна в область вывода окна и повторением запуска бегущей строки с начала, (мс)
InterCharacterWait ЗадержкаМеждуСимволами	Int	RW	Задержка между выводом символов для создания эффекта телетайпа, (мс)
BlinkInterval ИнтервалМигания	Int	RW	Период мигания символов (мс)
CharacterSet НаборСимволов	Int	RW	Текущий номер кодовой таблицы символов

Описание свойств

Data

Строка

В параметре указывается строка с данными, выводимыми в окно.

Допустимо использование следующих специальных символов: \$0D и \$0A. Если в тексте встречается последовательность этих двух символов, то осуществляется перевод на новую строку.

Окно сдвигается, если текущая строка последняя.

Attribute

Атрибут

В параметре указывается тип мерцания строки.

Если свойство CapBlink = 0, то значение Attribute игнорируется. Если свойство CapBlink = 1, то весь дисплей переводится в режим мигания, если есть хотя бы один символ с атрибутом «мигающий».

Если свойство CapBlink = 3, то мигают только символы с атрибутом «мигающий».

CurrentWindow

ТекущееОкно

[BX]

Свойство содержит номер текущего окна вывода текста.

CursorColumn

КурсорСтолбец

[BX]

Свойство задает номер столбца (абсциссу) позиции текущего окна, в которую будет выведен следующий символ. CursorColumn всегда должен находиться в диапазоне от 0 до Columns.

CursorRow

КурсорСтрока

[BX]

Свойство содержит номер строки (ординату) позиции текущего окна, в которую будет выведен следующий символ. CursorRow всегда должен находиться в диапазоне от 0 до Rows.

CursorUpdate

ОбновлятьКурсор

[BX]

- Если `CursorUpdate = TRUE`, то метод `DisplayText()` изменяет значения свойств `CursorColumn` и `CursorRow`, устанавливая их на позицию за последним выведенным символом.
- Если `CursorUpdate = FALSE`, то методы `DisplayText (Data, Attribute)` и `DisplayTextAT (Y, X, Data, Attribute)` значения свойств `CursorColumn` и `CursorRow` оставляют без изменений.

CharacterSet

НаборСимволов

[BX]

Свойство содержит номер текущей кодовой таблицы символов.

Кодовые таблицы символов смотрите в руководстве по эксплуатации дисплея.



Сдвига окна не происходит, если последний символ выводится в конце строки. В этом случае `CursorRow` устанавливается на эту строку, `CursorColumn` устанавливается в `Columns`. В результате необходимый сдвиг окна происходит только при следующем выводе текста в окно.

DisplayTextAT (Y, X, Data, Attribute) **ПоказатьТекстПоз (Y, X, Строка, Атрибут)**

Данный метод полностью идентичен методу `DisplayText (Data, Attribute)`, с той разницей, что первые два параметра определяют позицию в окне, с которой необходимо выводить строку.

Метод эквивалентен последовательности:

```
CursorColumn = X;
```

```
CursorRow = Y;
```

```
DisplayText (Data, Attribute);
```

Название	Тип	Дост.	Значения
Параметры			
Y Y	Int	RW	Номер строки положения курсора: 0 .. Rows
X X	Int	RW	Номер столбца положения курсора: 0 .. Columns
Data Строка	Str	RW	Строка, выводимая в окно

Название	Тип	Дост.	Значения
Attribute Атрибут	Int	RW	Тип мерцания: • 0 – мерцания нет; • 1 – аппаратное мерцание; • 2 – мерцание текста; • 3 – мерцание окна
Входные свойства			
CurrentWindow ТекущееОкно	Int	RW	Номер текущего окна
CursorUpdate ОбновлятьКурсор	Log	RW	Изменять значение свойств CursorColumn и CursorRow: FALSE / TRUE
MarqueeType ТипБегСтроки	Int	RW	Тип бегущей строки: • 0 – строка не запущена; • 1 – сдвиг строки вверх; • 2 – сдвиг строки вниз; • 3 – сдвиг строки влево; • 4 – сдвиг строки вправо
MarqueeFormat ФорматБегСтроки	Int	RW	Формат бегущей строки: • 1 – текст появляется по одному символу, затем перемещается к началу; • 2 – текст появляется сразу затем удаляется с экрана
MarqueeUnitWait ЗадержкаПоказаБегСтроки	Int	RW	Задержка между последовательными сдвигами столбцов и строк, (мс)
MarqueeRepeatWait ЗадержкаПовтораБегСтроки	Int	RW	Задержка между сдвигом последнего символа или строки окна в область вывода окна и повторением запуска бегущей строки с начала, (мс)
InterCharacterWait ЗадержкаМеждуСимволами	Int	RW	Задержка между выводом символов для создания эффекта телетайпа, (мс)
BlinkInterval ИнтервалМигания	Int	RW	Период мигания символов (мс)
CharacterSet НаборСимволов	Int	RW	Текущий номер кодовой таблицы символов

Описание свойств

Data

Строка

В параметре указывается строка с данными, выводимыми в окно.

Допустимо использование следующих специальных символов: \$0D и \$0A. Если в тексте встречается последовательность этих двух символов, то осуществляется перевод на новую строку.

Окно сдвигается, если текущая строка последняя.

Attribute

Атрибут

В параметре указывается тип мерцания строки.

- Если свойство CapBlink = 0, то значение Attribute игнорируется.
- Если свойство CapBlink = 1, то весь дисплей переводится в режим мигания, если есть хотя бы один символ с атрибутом «мигающий».
- Если свойство CapBlink = 3, то мигают только символы с атрибутом «мигающий».

CurrentWindow

ТекущееОкно

[BX]

Свойство содержит номер текущего окна вывода текста.

CursorUpdate

ОбновлятьКурсор

[BX]

- Если CursorUpdate = TRUE, то метод DisplayText() изменяет значения свойств CursorColumn и CursorRow, устанавливая их на позицию за последним выведенным символом.
- Если CursorUpdate = FALSE, то методы DisplayText (Data, Attribute) и DisplayTextAT (Y, X, Data, Attribute) значения свойств CursorColumn и CursorRow оставляют без изменений.

CharacterSet

НаборСимволов

[BX]

Свойство содержит номер текущей кодовой таблицы символов.

Кодовые таблицы символов смотрите в руководстве по эксплуатации дисплея.



При первом (после записи `DeviceEnabled = TRUE`) выводе на экран, если `Model = 1` и `DownloadFonts = TRUE`, драйвер загружает кириллический шрифт в дисплей.

RefreshWindow (Window) **ОбновитьОкно (Окно)**

Метод «перерисовывает» окно.

Данный метод делает окно с номером, переданным в параметре, текущим и обновляет его область вывода. Позиция курсора и координаты привязки области вывода не меняются.

Название	Тип	Дост.	Значения
Параметры			
Window Окно	Int	RW	Номер окна

ScrollText (Direction, Unit) **ПрокрутитьТекст (Направление, Сдвиг)**

Метод прокручивает текущее окно в заданном параметром `Direction` направлении.

Название	Тип	Дост.	Значения
Параметры			
Direction Направление	Int	RW	Направление сдвига окна: • 1 – вверх; • 2 – вниз; • 3 – влево; • 4 – вправо
Unit Сдвиг	Int	RW	Количество символов или строк, на которое прокручивать окно

ClearText () **ОчиститьТекст ()**

По этой команде драйвер уничтожает всю информацию в окне (заполняет его пробелами), а также устанавливает значения по умолчанию в свойствах: `MarqueeType`, `MarqueeFormat`, `MarqueeUnitWait`, `MarqueeRepeatWait`, `InterCharacterWait`, `BlinkInterval`.

Название	Тип	Дост.	Значения
----------	-----	-------	----------

Название	Тип	Дост.	Значения
Входные свойства			
CurrentWindow ТекущееОкно	Int	RW	Номер текущего окна
Выходные свойства			
MarqueeType ТипБегСтроки	Int	RW	Тип бегущей строки: 0 – бегущая строка не запущена
MarqueeFormat ФорматБегСтроки	Int	RW	Формат бегущей строки: 0
MarqueeUnitWait ЗадержкаПоказаБегСтроки	Int	RW	Задержка между последовательными сдвигами столбцов и строк, (мс): 0
MarqueeRepeatWait ЗадержкаПовтораБегСтроки	Int	RW	Задержка между сдвигом последнего символа или строки окна в область вывода окна и повторением запуска бегущей строки с начала, (мс): 0

Clear () **Очистить ()**

Метод очищает дисплей, но не окна.

Прочие методы

SelfTest () **СамоТестирование ()**

Метод запускает процедуру самотестирования дисплея.

Вывод дисплея из режима самотестирования (во время теста дисплей не принимает команд от ПК) возможен только после полного завершения теста и последующего выключения питания.

OpenDrawer () **ОткрытьЯщик ()**

Открывает денежный ящик.

Название	Тип	Дост.	Значения
Выходные свойства			

Название	Тип	Дост.	Значения
DrawerOpened ЯщикОткрыт	Log	RW	Состояние денежного ящика: FALSE / TRUE

Описание свойств

DrawerOpened

ЯщикОткрыт

[Вых]

- Если DrawerOpened = TRUE, то денежный ящик открыт.
- Если DrawerOpened = FALSE, то денежный ящик закрыт.

Ящик не откроется, если он заперт на ключ.

Поддерживаемое оборудование

Метод поддерживается только моделями с подключаемым денежным ящиком.

Подключение драйвера

1С: Предприятие v.7.7

Подключение драйвера происходит в глобальном модуле конфигурации для «1С: Предприятие» версии 7.7.

Перем Дисплей Экспорт; // Глоб. переменная для работы с драйвером

Процедура ПриНачалеРаботыСистемы()

// Загрузка внешней компоненты

Если ЗагрузитьВнешнююКомпоненту("Line1C.dll") <> 0 **Тогда**

// Создание объекта

Дисплей = **СоздатьОбъект**("AddIn.Line45");

Иначе

Сообщить("Внешняя компонента драйвера не найдена");

КонецЕсли;

КонецПроцедуры

Процедура ПриЗавершенииРаботыСистемы()

Дисплей = 0; // Отсоединение объекта

КонецПроцедуры

1С: Предприятие v.8.x

Подключение драйвера происходит в модуле приложения конфигурации для «1С: Предприятие» версии 8.x.

Перем Дисплей Экспорт; // Глоб. переменная для работы с драйвером

Процедура ПриНачалеРаботыСистемы()

Попытка

// Загрузка внешней компоненты

ЗагрузитьВнешнююКомпоненту("Line1C.dll");

Сообщить("Внешняя компонента загружена");

Попытка

// Создание объекта

Дисплей = **Новый**("AddIn.Line8");

Исключение

Сообщить("Объект не найден!");

КонецПопытки;

Исключение

Сообщить ("Внешняя компонента драйвера не найдена");

КонецПопытки;

КонецПроцедуры

Процедура ПриЗавершенииРаботыСистемы()

Дисплей = 0; // Отсоединение объекта

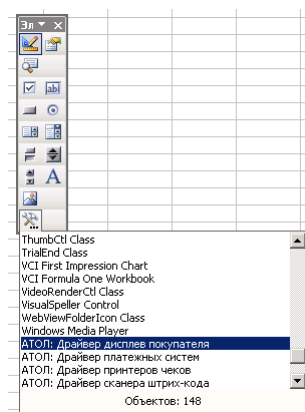
КонецПроцедуры

Microsoft VBA (Excel, Word и др.)

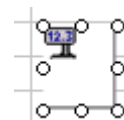
Откройте панель инструментов «Элементы управления».



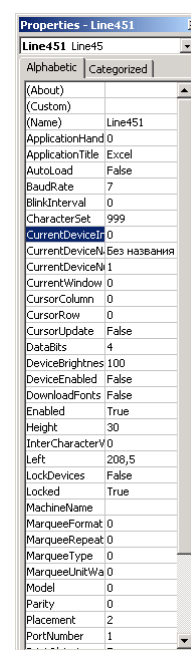
Нажмите кнопку «другие элементы» и выберите в списке «АТОЛ: Драйвер дисплеев покупателя этикеток v.8» либо «АТОЛ: Драйвер дисплеев покупателя этикеток v.8 (совместимый с v.6)».



Вставьте компонент на лист (форму).



Далее можно работать с драйвером, как с ActiveX компонентой. Через контекстное меню можно отобразить редактор свойств или визуальную страницу свойств драйвера.

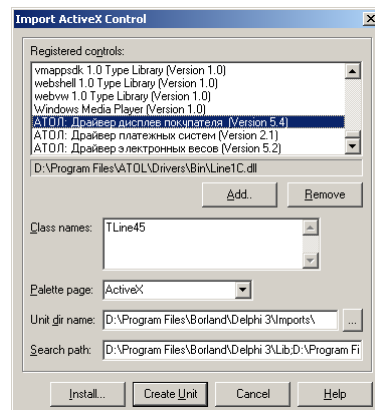


Borland Delphi, C++ Builder

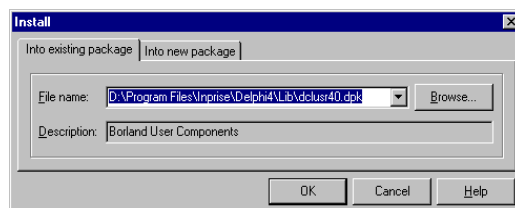
В среде ActiveX компоненту «АТОЛ: Драйвер дисплеев покупателя» можно поместить на палитру компонентов.

Выберите пункт «Import ActiveX Control...» в меню «Component»

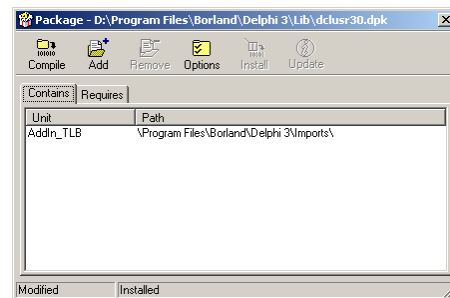
1. Выберите элемент «АТОЛ: Драйвер дисплеев покупателя v.8».
2. В поле «Palette Page» укажите страницу палитры, на которую желаете добавить компонент.
3. В поле «Unit dir name» задайте директорию, в которой будет сохранен импортируемый модуль.
4. Нажмите кнопку «Install...».



5. Укажите имя существующего (закладка «Into existing package») или нового (закладка «Into new package») пакетного файла.



6. Нажмите «Yes» для подтверждения перекомпиляции пакетного файла.
7. Нажмите «OK» в окне сообщения об удачной перекомпиляции.
8. Закройте окно пакетного файла.
9. Нажмите «Yes» для подтверждения сохранения пакетного файла.



После этого на закладке «ActiveX» будут расположены компоненты «TLine45» и «TLine8».

Приложение 1. Коды и описание ошибок

Код ошибки	Описание
0	Ошибок нет
-1	Нет связи
-2	Операция зарезервирована для будущих версий
-3	Порт недоступен
-4	Ключ защиты не найден
-5	Работа драйвера прервана пользователем
-6	Недопустимое значение
-7	Невозможно добавить устройство
-8	Невозможно удалить устройство
-9	Устройство не найдено
-10	Неправильная последовательность операций
-11	Устройство не включено
-12	Не поддерживается в данной версии оборудования
-13	Драйвер не смог загрузить необходимые модули
-14	Порт занят приложением: (\\<имя ПК>\<описание приложения-клиента>\<описание драйвера>)
-15..-198	Зарезервировано
...	...
-198	Зарезервировано
-199	Неопознанная ошибка

+7(495) 730-7420
www.atol.ru



Исключительные права
на программное обеспечение
и документацию принадлежат
компании АТОЛ