Star Micronics TSP100 FuturePRNT Software Development Kit

The Star Micronics TSP100 FuturePRNT Software Development Kit allows developers to intergrate all of the most advanced features of the TSP100 futurePRNT into their own applications. All of the specific features of the TSP100 futurePRNT are available as individual Active X components which can easily be embedded into the developer's new or existing applications.

Overview

The Star Micronics TSP100 SDK is provided as a set of ActiveX components. ActiveX components can be used by most Windows based development environments, including Visual Basic 5/6, Visual Basic for Applications, C/C++, Access, Delphi, .NET based languages and more.

Each component may ve used visibly, providing a ready-made interface to the user of your application, or invisibly, allowing your application to modify the behaviour of your printer directly.

All configuration components are held together by a central 'ConfigurationHandler' component. The ConfigurationHandler manages loading/saving and reading/writing into your printer configuration, most configuration components must be assigned a configuration handler (via the 'SetConfigurationHandler()' method) before they can be used.
Multiple instances of the ConfigurationHandler should not be allowed to access a single configuration file at the same time, however it is safe to assign a single ConfigurationHandler to many other components.

Only one of each component type should have access to a single ConfigurationHandler instance. To assign a single ConfigurationHandler to multiple instances of the same object type may give unpredictable results or cause corruption to your printer configuration. The exception to this rule is the GeneralSettings component, it is safe to have several GeneralSettingc components as long as each one displays unique settings.

User Controls (Public Functions/Properties)

AddImages 

The AddImages component is used to add pre-imported images to the top and/or bottom of your printed pages. Images should have already been imported and saved into your printer configuration using the 'SelectImages' component.
The AddImages component has a complete user interface which may be left visible for use by end users of your software, otherwise, the component may be made invisible and manipulated using it's API.

Functions:

SetConfigurationHandler(ch As ConfigurationHandler)
Specify the ConfigurationHandler object that the AddImages object will use for access to the configuration file. A configurationHandler myst be assigned to your AddImages objct, and the a configuration must be loaded before the AddImaged object may be used.

AddTopImage(imageName As String, imageAlign As String)
Add a new image that will be added to the top of all receipts.
0 left, image will be left justified on the receipt.
1 centre, image will be centred on the receipt.
2 right, image will be right justified on the receipt.

AddTopImageDialog()
Add a new image. This function will open a dialog, prompting the user to select an image that will be added to the top of all receipts.

AddBottomImage(imageName As String, imageAlign As String)
Add a new image that will be added to the bottom of all receipts.
0 left, image will be left justified on the receipt.
1 centre, image will be centred on the receipt.
2 right, image will be right justified on the receipt.

AddBottomImageDialog()
Add a new image. This function will open a dialog, prompting the user to select an image that will be added to the bottom of all receipts.

DeleteTopImage(imageName As String)
Prevent the named image from printing at the top of all documents.

DeleteBottomImage(imageName As String)
Prevent the named image from printing at the bottom of all documents.

SelectMergeImage()
Display a dialog, allowing the user to select a merge image.

ClearMergeImage()
Disable image merging.

GetTopImageName(Index As Integer) As String
Return the name of the top image at the given index position.

GetBottomImageName(Index As Integer) As String
Return the name of the bottom image at the given index position.

Properties:

MergeImage (String)
Retreive or set the name of the current merge image.

TopPrintMode (Integer)
Retreive or set the top image print mode:
0 Print All, every top image will print on every printed page.
1 Cycle, only one image will print on each page. If multiple images have been specified then the image will be changed on each receipt in sequence.

BottomPrintMode (Integer)
Retreive or set the top image print mode:
0 Print All, every top image will print on every printed page.
1 Cycle, only one image will print on each page. If multiple images have been specified then the image will be changed on each receipt in sequence.

TopImageListCount (Integer) (Read Only)
Retrieve the number of images in the top image list.

BottomImageListCount (Integer) (Read Only)
Retrieve the number of images in the bottom image list.

ConfigurationHandler

The ConfigurationHandler is used to access printer configurations directly.
Most of the ConfigurationHandler methods are undocumented here and should not be used directly, but are used by the other SDK components.
All components that read or write into the printer configuration require access to a ConfigurationHandler component.
When writing an application, the correct procedure for setting-up components is:
1. Instantiate all of your components (this is automatic if they have been drawn directly onto a form)
2. Assign your ConfigurationHandler object to each object that requires it.
3. Load a configuration into your ConfigurationHandler

Functions:

LoadConfigurationFromXML(xmlfilename As String) As Boolean
Load a configuration file specified by the “xmlfilename” parameter into the configuration handler. Any components that are set to use this ConfigurationHandler object will modify the configuration file specified.

ImportConfigurationToXML(xmlfilename As String)

SaveConfigurationToXML(xmlfilename As String) As Boolean
Save all configuration changes made to the configuration handler to a configuration file specified using the “xmlfilename” parameter.

LoadConfigurationFromQueue(QueueName As String) As Boolean
Load the configuration file currently being used by the specified printer queue name into the ConfigurationHandler. Any components that are set to use this ConfigurationHandler object will modify said configuration file.

SaveConfigurationToQueue(QueueName As String) As Boolean
Save all configuration changes made to the configuration handler to a configuration file specified using the “xmlfilename” parameter.

Properties:

PrinterQueueName (String) (Read Only)
Retrieve the printer queue name that this ConfigurationHandler is currently set to use to determine which configuration file to load and save.

printerSN (String) (Read Only)
Return the printers USB serial number.

Crop 

The Crop component enables you to specify a cropping length from the beginning and/or end of any printed pages. This may be useful for hiding outdated headers or footers from receipts (which may be replaced using the AddImages component.

Functions:

SetConfigurationHandler(ch As ConfigurationHandler)
Specify the ConfigurationHandler object that the AddImages object will use for access to the configuration file. A configurationHandler myst be assigned to your AddImages objct, and the a configuration must be loaded before the AddImaged object may be used.

Properties:

TopCrop (Single)
Specify the top of page crop length in millimeters. The specified length will be removed from the top of all subsequent printed pages.
The top crop length is accurate to the nearest 1/8mm.

BottomCrop (Single)
Specify the bottom of page crop length in millimeters. The specified length will be removed from the bottom of all subsequent printed pages.
The bottom crop length is accurate to the nearest 1/8mm.

EnumPrinters 

The EnumPrinters component is used to generate a list of all printers on the users system. It may optionally show only devices thar are compatible with this SDK, or all printers. This component does not require a ConfigurationHandler as it does not directly read or write into a printer configuration.
It is typically used to determine the queue name of a printer to configure.

Functions:

RefreshPrinterList()
Force the internal printer list to be updated. This may be useful if your software can cause printers to be added, deleted or renamed, if if it is possible that a separate application (or the user) may have done so since the object was instantiated.

GetPrinterName(Index As Integer) As String
Retreive printer queue name from the list at the supplied index.

Properties:

ShowAll (Boolean)
If this property is true then all printers that are installed on the system will be listed. If it is false then onld devices that are compatible with this SDK will be listed.

ListCount (Integer)(Read Only)
Return the number of devices listed.

SelectedPrinterName (String)

If the object is visible, then users are able to manually select a printer from the list and this property will get the currently selected printer queue name. If no printer has been selected then this will return an empty string ("").

Events:

Changed(QueueName As String)

If this component is visible, the Changed event will be triggered when the user clicks on a printer on the list. The selected printer queue name is provided by the QueueName string.

GeneralSettings

The GeneralSettings component provides access to many different common POS printer configuration options such as the default printer character set. It is safe to have more than one GeneralSettings object linked to a single ConfigurationHandler as long es each GeneralSettings object is used to manipulate different options.
If the object is visible, then it will present a list of options to the user, it is possible to modify the contents of this list by changing property values.

Functions:

SetConfigurationHandler(ch As ConfigurationHandler)

Specify the ConfigurationHandler object that the GeneralSettings object will use for access to the configuration file. A configurationHandler must be assigned to your GeneralSettings objct, and the a configuration must be loaded before the GeneralSettings object may be used.

Properties:

RowSpacing (Long)

Specify the vertical spacing (in twips) between listed options when the object is visible.

labelAlignment (AlignmentConstants)

Specify the alignment of the list abels then the object is visible.

ShowZeroStyle (Boolean)

Specify whether to show the 'Zero Style' list option, if the GeneralSettings object is visible.

ShowInternationalCharacterSet (Boolean)

Specify whether to show the 'International Character Set' list option, if the GeneralSettings object is visible.

ShowCodePage (Boolean)

Specify whether to show the 'Code Page' list option, if the GeneralSettings object is visible.

ShowCharacterEncoding (Boolean)

Specify whether to show the 'Character Encoding' list option, if the GeneralSettings object is visible.

ShowLineSpacing (Boolean)

Specify whether to show the 'Line Spacing' list option, if the GeneralSettings object is visible.

ShowPrintDensity (Boolean)

Specify whether to show the 'Print Density' list option, if the GeneralSettings object is visible.

ShowPrintSpeed (Boolean)
Specify whether to show the ‘Print Speed’ list option, if the GeneralSettings object is visible.

ShowPrintWidth (Boolean)
Specify whether to show the ‘Print Width’ list option, if the GeneralSettings object is visible.

ShowITFBearerBars (Boolean)
Specify whether to show the ‘ITF barcode Bearer Bars’ list option, if the GeneralSettings object is visible.

ShowUPCAExtendGuardBars (Boolean)
Specify whether to show the ‘UPC-A Barcode Extender Guard Bars’ list option, if the GeneralSettings object is visible.

ShowEAN13ExtendGuardBars (Boolean)
Specify whether to show the ‘EAN-13 Barcode Extended Guard Bars’ list option, if the GeneralSettings object is visible.

ShowCODE39ExtendGuardBars (Boolean)
Specify whether to show the ‘CODE 39 Barcode Extended Guard Bars’ llist option, if the GeneralSettings object is visible.

ZeroStyle (EZeroStyle)
Get or set the method used to draw zero characters when using the printers standard font. Possible values are:
ZeroStyle_Standard
ZeroStyle_Slashed

InternationalCharacterSet (EInternationalCharacterSet)
Get or set the default International Character Set when using the printers standard font. Possible values are:
InternationalCharacterSet_USA
InternationalCharacterSet_France
InternationalCharacterSet_Germany
InternationalCharacterSet_uk
InternationalCharacterSet_Denmark_1
InternationalCharacterSet_Sweden
InternationalCharacterSet_Italy
InternationalCharacterSet_Spain_1
InternationalCharacterSet_Japan
InternationalCharacterSet_Norway
InternationalCharacterSet_Denmark_2
InternationalCharacterSet_Spain_2
InternationalCharacterSet_LatinAmerica
InternationalCharacterSet_Korea
InternationalCharacterSet_Ireland
InternationalCharacterSet_Legal

CodePage (ECodePage)
Specify the default code page when using the printers standard font and Single Byte character endoding. Possible values are:
CodePage_Normal
CodePage_437
CodePage_Katakana

CodePage_858
CodePage_852
CodePage_860
CodePage_861
CodePage_863
CodePage_865
CodePage_866
CodePage_855
CodePage_857
CodePage_862
CodePage_864
CodePage_737
CodePage_851
CodePage_869
CodePage_928
CodePage_772
CodePage_774
CodePage_874
CodePage_1252
CodePage_1250
CodePage_1251
CodePage_3840
CodePage_3841
CodePage_3843
CodePage_3844
CodePage_3845
CodePage_3846
CodePage_3847
CodePage_3848
CodePage_1001
CodePage_2001
CodePage_3001
CodePage_3002
CodePage_3011
CodePage_3012
CodePage_3021
CodePage_3041
CodePage_Blank

CharacterEncoding (ECharacterEncoding)

Specify the default character encoding when using the printers standard font. Possible values are:

CharacterEncoding_SingleByte
CharacterEncoding_Japanese
CharacterEncoding_Chinese
CharacterEncoding_Taiwanese
CharacterEncoding_Korean

LineSpacing (ELineSpacing)

Specify the default line spacing when using the printers standard font. Possible values are:

LineSpacing_4mm
LineSpacing_3mm

PrintDensity (EPrintDensity)

Specify or retrieve the printers default print density. Density can be adjusted to improve print quality on under or over sensitive paper stock.

PrintDensity_130 = 0

PrintDensity_120
PrintDensity_110
PrintDensity_100
PrintDensity_90
PrintDensity_80
PrintDensity_70

PrintSpeed (EPrintSpeed)
Specify or retrieve the printers default print speed. Speed can be lowered to improve print quality on under sensitive paper stock. Possible values are:
PrintSpeed_High
PrintSpeed_Medium
PrintSpeed_Low

printWidth (EPrintWidth)
Specify or retrieve the printers paper width setting. Possible values are:
PrintWidth_72mm
PrintWidth_51mm

Journal

The Journal component can be used to enable or disable electronic journaling. When Journalling is enabled, all print jobs will be stored locally for reference.
If this component is visible then it will be presented as a checkbox that the user may use to enable or disable journaling.

Functions:
SetConfigurationHandler(ch As ConfigurationHandler)
Specify the ConfigurationHandler object that the Journal object will use for access to the configuration file. A configurationHandler must be assigned to your Journal objct, and the a configuration must be loaded before the Journal object may be used.

Properties:
JournalEnabled (Boolean)
Enable or disable electronic yournalling.

SelectImages

The SelectImages component is used to import images into the central image list. The image list is used as a source of images to be used with other printer features such as the AddImages component (to add automatic top of page, bottom of page, or merge images) or the coupon printing feature.
In the current software release (1.2) This component can only be used visibly with user interaction.

Functions:
SetConfigurationHandler(ch As ConfigurationHandler)
Specify the ConfigurationHandler object that the SelectImages object will use for access to the configuration file. A configurationHandler must be assigned to your SelectImages, and the a configuration must be loaded before the SelectImages l object may be used.

TextSearch

The TextSearch component allows you to set-up text based triggers which may be used to modify your printes deceipts. Each trigger has a text key associated with it, and a set of

options that may be used to modify the printed page when this text trigger is found on that page.

Functions:

SetConfigurationHandler(ch As ConfigurationHandler)

Specify the ConfigurationHandler object that the SelectImages object will use for access to the configuration file. A configurationHandler must be assigned to your SelectImages, and the a configuration must be loaded before the SelectImages l object may be used.

GetEndingModeState(endingType As PageEndingTypes) As Boolean

Retreive the current state of the page ending method specified by ending|Type. Possible values for endingType are:

CutEnding = 0
FormFeedEnding = 1
LineFeedEnding = 2

Each of these specify a method by which the printer software will detect the end of a page. It is required that many features (such as Journal, Preview, coupon printing and Add Images) can correctly detect the end of a printed page. You may use this method (and the related SetEndingModeState()) to choose the method by which page ends are detected.

CutEnding – when enabled a page end will be detected whenever a cut or feed-to-tearbar command is sent to the printer. This is true by default on new installations.
FormFeedEnding – when enabled, a page end will be detected when a form feed command is sent to the printer. This is true by default on new installations
LineFeedEnding – when enabled, a page end will be detected when a consecutive group of line feeds is detected. The number of line feeds required to trigger a page end can be set using the EndPageLineFeeds property. This is false by default on new installations.

SetEndingModeState(endingType As PageEndingTypes, state As Boolean)

Set the state of each page end detection method. Refer to GetEndingModeState() above for more information.

AddTextKey()

Add a new text key, this will display a dialog prompting the user to enter the text for the new text key.

DelTextKey(KeyName As String)

Delete the key whose search text matches KeyName.

GetKeyName(Index As Integer) As String

Return the key name at the given index

Properties:

ReplaceSelected (Boolean)

Enable or disable text replacement when the currently selected key is found on a page. Use the SelectedKeyName property to specify the key that you are configuring and the ReplaceText property to check or set the replacement text string.

ReplaceText (String)

Retreive or set the replacement text for the currently selected search key.

MergeSelected (Boolean)

Enable or disable modification of the page merge settings when the current key is found. If it is enabled, then the page merge settings will be changed (either to disable merging of to change the image) when this key is found.

MergeChangeMode (MergeModes)
If the MergeSelected property is true for the currently selected text key then this property can be used to specify exactly how the key will affect the merge settings when it is found on a page. Possible values are:
MergeMode_Disablemerging – this indicates thet the merging will be disabled if the currently selected key is found on a page. This may be useful when printing barcodes as it is possible that a merge image will make a barcode difficult to scan.
MergeMode_ChangeMergeImage – indicates that the merge image should be changed if this text key is found on a page. This feature can be used to create themed receipts. Use the MergeImage propert to retrieve or set the name of the image that will be used.

MergeImage (String)
Retreive or set the name of the image that will be merged with a printed page if the currently selected text key is found on it. This only has effect if MergeSelected = True and MergeChangeMode = MergeMode_ChangeMergeImage.

CouponsSelected (Boolean)
Enable or disable modification of the page coupon settings when the current key is found.

CouponChangeMode (CouponModes)
If the CouponSelected property is true for the currently selected text key then this property can be used to specify exactly how the key will affect the coupon printing settings when it is found on a page. Possible values are:
CouponMode_DisableCoupons– this indicates thet coupon printing will be disabled if the currently selected key is found on a page.
CouponMode_AddCoupon – indicates that a coupon will be added if the currently selected key is found on a page.

CouponImage (String)
If CouponChangeMode = CouponMode_AddCoupon then the CouponImage property can be used to retrieve or set the coupon image to be printed.

SelectedKeyName (String)
Retreive or set the currently selected text key. The value of this property may or may not affect the configuration of this control when certain functions are executed.

KeyNameCount (Integer) (Read Only)
Retreive the amount of text key names currently listed in this control.

EndPageLineFeeds (Integer)
Retreive or set the number of line feeds the printer will interpret as the end of a print job if the controls Ending Mode State is set to “LineFeedEnding”

WallMount

Functions:
SetConfigurationHandler(ch As ConfigurationHandler)
Specify the ConfigurationHandler object that the WallMount object will use for access to the configuration file. A ConfigurationHandler must be assigned to your WallMount

control, and the configuration file must be loaded before the WallMount object may be used.

Properties:

WallMountEnabled (Boolean)

Enable or disable wallmount printing (reverse text and images).

CashDrawerKicker

This component allows for direct control of any Cash Drawer devices that are connected to the TSP100. Specifically, this control has been designed so that Cash Drawers can be opened *without* needing to print anything.

Functions:

OpenDrawer() (Error on Failure)

Opens the Cash Drawer in the manner specified in the properties of the control. The potential errors of this function are as follows:

1 + (vbObjectError + 512) – indicating that the PrinterQueueName property does is not associated with a valid TSP100 device

2 + (vbObjectError + 512) – indicating that a communications error occurred while trying to control the TSP100 device

WaitForPrimaryCashDrawerSignalLevel(SignalLevel As Boolean, TimeoutMilliseconds As Long) (Error on Failure)

Polls the Primary Cash Drawer's status until it begins signaling at the specified signal level or will timeout after the specified timeout period (1 second = 1 * 1000 milliseconds). The potential errors of this function are as follows:

1 + (vbObjectError + 512) – indicating that the PrinterQueueName property does is not associated with a valid TSP100 device

2 + (vbObjectError + 512) – indicating that a communications error occurred while trying to control the TSP100 device

3 + (vbObjectError + 512) – indicating that the control timed-out before the Cash Drawer attained the specified signal level

Properties:

PrinterQueueName (String)

This property holds the printer queue name (as established in the Windows Printer's & Faxes folder) associated with the device being controlled.

CashDrawerCircuit (CashDrawerCircuit)

This property controls which Cash Drawer is opened during subsequent calls to the OpenDrawer method. Possible values are:

PrimaryCircuit

SecondaryCircuit

EnergizingPulseWidth (Integer)

Retreives or sets the width (in milliseconds) of the energizing pulse that the printer will send to the Cash Drawer when opening it during subsequent calls to the OpenDrawer method. This property is made available to accommodate the many different types of Cash Drawers available on the market, some of which require only a short engergizing pulse and others which require longer energizing pulses. This parameter can be set to any Integer value between 10 and 1270 (meaning 10 milliseconds and 1270 milliseconds respectively). This property is only applicable to the primary Cash Drawer.

DelayPulseWidth (Integer)

Retreives or sets the width (in milliseconds) of the delay pulse that the printer will wait for after opening the Cash Drawer during subsequent calls to the OpenDrawer method. This parameter can be set to any Integer value between 10 and 1270 (meaning 10 milliseconds and 1270 milliseconds respectively). This property is only applicable to the primary Cash Drawer.

IsPrimaryCashDrawerSignalling (Boolean)(Read Only)(Error on Failure)

Retreives the current (real-time) status of the Primary Cash Drawer. Specifically, it returns true if the Primary Cash Drawer is outputting a voltage level to the printer and false if no voltage level is being outputted. Typically cash drawers output voltage to the printer when they are opened, but some cash drawers have negatively-wired detection switches, and so this method simply indicates the status of the drawer's outputted voltage; this status must be paired with knowledge of the specific Cash Drawer being used to determine if the drawer is opened or closed. The potential errors of this property are as follows:

1 + (vbObjectError + 512) – indicating that the PrinterQueueName property is not associated with a valid TSP100 device

2 + (vbObjectError + 512) – indicating that a communications error occurred while trying to control the TSP100 device