# 1 Content

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Content</td>
</tr>
<tr>
<td>1.1</td>
<td>Table of Figures</td>
</tr>
<tr>
<td>2</td>
<td>Disclaimer</td>
</tr>
<tr>
<td>3</td>
<td>General</td>
</tr>
<tr>
<td>3.1</td>
<td>TBarCode OCX - a Microsoft® ActiveX® compliant Barcode Control</td>
</tr>
<tr>
<td>3.2</td>
<td>Scope of this document</td>
</tr>
<tr>
<td>3.3</td>
<td>Installation</td>
</tr>
<tr>
<td>3.4</td>
<td>Registering the Control</td>
</tr>
<tr>
<td>3.5</td>
<td>Licensing</td>
</tr>
<tr>
<td>3.5.1</td>
<td>Ordering a license</td>
</tr>
<tr>
<td>3.6</td>
<td>Redistribution as part of your Application</td>
</tr>
<tr>
<td>3.7</td>
<td>Limitations of the Demo Download</td>
</tr>
<tr>
<td>3.8</td>
<td>Sample Applications</td>
</tr>
<tr>
<td>3.9</td>
<td>Product Documentation</td>
</tr>
<tr>
<td>4</td>
<td>Terms and Definitions</td>
</tr>
<tr>
<td>4.1</td>
<td>ActiveX</td>
</tr>
<tr>
<td>4.2</td>
<td>Container Application</td>
</tr>
<tr>
<td>4.3</td>
<td>Symbology (Barcode Type)</td>
</tr>
<tr>
<td>4.4</td>
<td>Encoding</td>
</tr>
<tr>
<td>4.5</td>
<td>Data Binding</td>
</tr>
<tr>
<td>5</td>
<td>Use in Common Applications</td>
</tr>
<tr>
<td>5.1</td>
<td>General</td>
</tr>
<tr>
<td>5.1.1</td>
<td>Inserting an ActiveX Control</td>
</tr>
<tr>
<td>6</td>
<td>Use in Microsoft Office Applications</td>
</tr>
<tr>
<td>6.1</td>
<td>Barcodes in Microsoft Word</td>
</tr>
<tr>
<td>6.1.1</td>
<td>Insert TBarCode into a document</td>
</tr>
<tr>
<td>6.1.2</td>
<td>Insert TBarCode for Scripting with VBA (Visual Basic for Applications)</td>
</tr>
<tr>
<td>6.1.3</td>
<td>Insert TBarCode per Code (VBA)</td>
</tr>
<tr>
<td>6.1.4</td>
<td>Mail Merge Documents</td>
</tr>
<tr>
<td>6.2</td>
<td>Barcodes in Microsoft Excel</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Insert TBarCode into a spreadsheet</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Use TBarCode with VBA (Visual Basic for Applications)</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Linked Cell (Data Binding)</td>
</tr>
<tr>
<td>6.2.4</td>
<td>Insert TBarCode per Code (VBA)</td>
</tr>
<tr>
<td>6.3</td>
<td>Barcodes in Microsoft Access</td>
</tr>
<tr>
<td>6.3.1</td>
<td>Insert TBarCode into a Microsoft Access Database or Report</td>
</tr>
<tr>
<td>6.3.2</td>
<td>VBA (Visual Basic for Applications)</td>
</tr>
<tr>
<td>6.3.3</td>
<td>Data Binding</td>
</tr>
<tr>
<td>6.4</td>
<td>Barcodes in Microsoft InfoPath®</td>
</tr>
<tr>
<td>6.4.1</td>
<td>Data Binding</td>
</tr>
<tr>
<td>6.4.2</td>
<td>Prepare Microsoft InfoPath to use TBarCode OCX</td>
</tr>
<tr>
<td>6.4.3</td>
<td>Insert TBarCode OCX into a Microsoft InfoPath Document</td>
</tr>
<tr>
<td>6.4.4</td>
<td>Change Barcode Settings in a Microsoft InfoPath Document</td>
</tr>
<tr>
<td>7</td>
<td>Use in Development-Environments (IDE)</td>
</tr>
<tr>
<td>7.1</td>
<td>Microsoft Visual Basic</td>
</tr>
<tr>
<td>7.1.1</td>
<td>Use TBarCode in a Visual Basic Project</td>
</tr>
<tr>
<td>7.2</td>
<td>Microsoft Visual C++ 6.0</td>
</tr>
<tr>
<td>7.2.1</td>
<td>Use TBarCode in a MFC Application</td>
</tr>
<tr>
<td>7.2.2</td>
<td>Use TBarCode in a non MFC Application</td>
</tr>
<tr>
<td>7.3</td>
<td>Microsoft Visual Studio .NET 2003</td>
</tr>
<tr>
<td>7.3.1</td>
<td>Use TBarCode as Control</td>
</tr>
<tr>
<td>7.3.2</td>
<td>Use TBarCode as Library</td>
</tr>
<tr>
<td>7.4</td>
<td>Microsoft Visual Studio .NET 2005</td>
</tr>
<tr>
<td>7.4.1</td>
<td>Use TBarCode as Control</td>
</tr>
<tr>
<td>7.4.2</td>
<td>Use TBarCode as Library</td>
</tr>
<tr>
<td>7.5</td>
<td>Microsoft Visual FoxPro</td>
</tr>
<tr>
<td>7.5.1</td>
<td>Inserting TBarCode into a FoxPro-Project</td>
</tr>
<tr>
<td>7.5.2</td>
<td>Reports &amp; Labels</td>
</tr>
<tr>
<td>7.6</td>
<td>Microsoft Navision®</td>
</tr>
<tr>
<td>7.7</td>
<td>Microsoft Office / VBA</td>
</tr>
<tr>
<td>7.8</td>
<td>Borland Delphi / Borland C++ Builder</td>
</tr>
<tr>
<td>7.8.1</td>
<td>Using TBarCode in Borland Delphi 6/7</td>
</tr>
<tr>
<td>7.8.2</td>
<td>Using TBarCode in Borland Delphi 2005 (Delphi for Win32)</td>
</tr>
<tr>
<td>7.8.2.1</td>
<td>Create a package for TBarCode OCX</td>
</tr>
<tr>
<td>7.8.2.2</td>
<td>Saving the packages</td>
</tr>
</tbody>
</table>
1.1 Table of Figures

Figure 1: Using TBarCode in Microsoft Word (1) 11
Figure 2: Using TBarCode in Microsoft Word (2) 12
Figure 3: Using TBarCode in Microsoft Word with VBA 13
Figure 4: Using TBarCode in Microsoft Excel 15
Figure 5: Using TBarCode in Microsoft Excel with VBA 16
Figure 6: Using TBarCode in Microsoft Excel (Data Binding) 17
Figure 7: Insert TBarCode per Code (VBA) 18
Figure 8: Using TBarCode in Microsoft Access 19
Figure 9: Using TBarCode in Microsoft Access (Data Binding) 20
Figure 10: Using TBarCode in Microsoft InfoPath (1) 22
Figure 11: Using TBarCode in Microsoft InfoPath (2) 23
Figure 12: Using TBarCode in Microsoft InfoPath (3) 24
Figure 13: Using TBarCode in Microsoft InfoPath (4) 24
Figure 14: Insert TBarCode into a Microsoft InfoPath Document 25
Figure 15: Changing TBarCode Settings in a Microsoft InfoPath Document 26
Figure 16: Using TBarCode in MSVC++ 6.0 (MFC Application) (1) 27
Figure 17: Using TBarCode in MSVC++ 6.0 (MFC Application) (2) 28
Figure 18: Using TBarCode in MSVC++ 6.0 (MFC Application) (3) 28
Figure 19: Using TBarCode in Visual Studio .Net 2003 (as Control) 29
Figure 20: Using TBarCode in Visual Studio .Net 2003 (as Library) 30
Figure 21: Using TBarCode in Visual Studio .Net 2005 (as Control) 31
Figure 22: Inserting TBarCode in PowerBuilder 39
Figure 23: PowerBuilder DataWindow “Display As Picture” 40
Figure 24: PowerBuilder DataWindow Bitmap function 40
2 Disclaimer

The actual version of this product (document) is available as is. TEC-IT declines all warranties which go beyond applicable rights. The licensee (or reader) bears all risks that might take place during the use of the system (the documentation). TEC-IT and its contractual partner cannot be penalized for direct and indirect damages or losses (this includes non-restrictive, damages through loss of revenues, constriction in the exercise of business, loss of business information or any kind of commercial loss), which is caused by use or inability to use the product (documentation), although the possibility of such damage was pointed out by TEC-IT.

We reserve all rights to this document and the information contained therein. Reproduction, use or disclosure to third parties without express authority is strictly forbidden.

Für dieses Dokument und den darin dargestellten Gegenstand behalten wir uns alle Rechte vor. Vervielfältigung, Bekanntgabe an Dritte oder Verwendung außerhalb des vereinbarten Zweckes sind nicht gestattet.

© 1998-2006
TEC-IT Datenverarbeitung GmbH
Wagnerstr. 6
A-4400 Austria
t.: +43 (0)7252 72720
f.: +43 (0)7252 72720 77
http://www.tec-it.com
3 General

3.1 TBarCode OCX - a Microsoft® ActiveX® compliant Barcode Control

TBarCode OCX is a full-featured barcode generator with a broad range of application areas. TBarCode OCX complies with Microsoft's ActiveX specification and can therefore be used in a bunch of well-known applications (like word-processors or development environments). It is very easy to use - embedding a barcode is done with a few mouse-clicks only.

All barcodes in common use are supported – ranging from linear barcodes (like Code2of5, Code39, Code128, EAN UPC, Postcodes, …) to high-density 2D barcodes (like PDF417, MaxiCode, Data Matrix, QR-Code…) and the UCC/EAN Composite Symbologies.

3.2 Scope of this document

This tutorial concentrates on the first steps for using TBarCode. We show you how to embed the barcode control into Microsoft Office¹ and common Development Environments as well as various other applications.

3.3 Installation

Download TBarCode OCX (http://www.tec-it.com/Download Bar Code Software for Windows). Execute the downloaded file to install the software. The setup-application copies all files to your computer and registers² the ActiveX control automatically within the Microsoft® Windows® operating system.

After the installation process the ActiveX Control TBarCode can be inserted into the desired application (see Use in Common Applications).

3.4 Registering the Control

The ActiveX control must be registered within the Windows Operating System prior use. This registration is performed automatically by the installation (the setup program).

To learn more about licensing (unlocking the demo version) read ahead in the next chapter.

If you don't use the setup provided by TEC-IT, you can register the ActiveX control manually with the "REGSVR32.EXE" utility provided by Microsoft (included in your Windows operating system or your development environment). Without this kind of registration you can not use ActiveX Controls.

¹ In this document Microsoft Office 2003 is used for demonstration purposes. But TBarCode can be used with previous (and upcoming) Office versions, too.
² Registering in this context means to „introduce“ the component to the operating system. This is done by making special entries into the Windows Registry - this is a common procedure for all ActiveX Controls.
3.5 Licensing

TBarCode is available in different license types.

- A **Single** license for the use of the software on exactly one computer (one concurrent user).
- A **Site** license for the use of the Software on several computers within one Site of the licensee, however limited to a maximum of 250 concurrent installations or 250 concurrent users.
- A **Developer** license for the redistribution of the Software as part of the Licensee’s software application(s). The number of redistributions per license is limited to 10000 installations or to 10000 users. Each developer building applications with the Software needs one Developer License.
- A **Web** license for the use of the Software on one server for internet-, intranet and other http-based applications. The number of users per license is limited to 10000. Each web-server instance needs one Web-License.

For users of TBarCode OCX: the ConvertToStream functions are licensed only with a **web** license.

- We spoke about “registering” in the previous chapter. This kind of “registration” is not the same as licensing. “Registering” a component makes the control available within the Windows environment. “Licensing” authorizes you to use the product. You can not use an ActiveX component without registration, but you can use it for testing purposes without licensing.
- To learn about the restrictions in the demo mode, please refer to [Limitations of the Demo Download](#).

3.5.1 Ordering a license

You can order a license key online from [http://www.tec-it.com/orderform](http://www.tec-it.com/orderform)

Or contact [sales@tec-it.com](mailto:sales@tec-it.com) if you have questions.

3.6 Redistribution as part of your Application

If you build applications with TBarCode, which are distributed to your customers, you need a so-called Developer license.

Without registering the component file (*.OCX) you can not use an ActiveX control. The .OCX file contains the ActiveX Type Library.

So make sure that the TBarCode component is registered on each target machine. If you use an installer this can be achieved by marking the .OCX file as "self-registering". If you don’t use an installer, you have to register the .OCX file with the system tool "REGSVR32.EXE".

3.7 Limitations of the Demo Download

TBarCode creates all barcodes correctly. In the unlicensed mode each barcode is marked with a small demo text printed over it. In most cases, except for very small barcodes, this text does not influence the readability of the barcode. If you want to test the product without the demo text, we can send you a temporary license key (please contact [sales@tec-it.com](mailto:sales@tec-it.com)).

- After a period of 90 days (after installation) the barcode generator can be deactivated automatically if it has not been licensed by the user. Applying a license key from TEC-IT will activate the product in an unlimited manner.
3.8 Sample Applications


- Note: Install TBarCode prior to use of the sample applications. TBarCode OCX (the ActiveX version) or TBarCode DLL must be installed before you can run the samples!

3.9 Product Documentation


For barcode specific information we offer a document “Barcode Reference”, which can be also downloaded from the link above.
4 Terms and Definitions

4.1 ActiveX

An ActiveX control is an object, which can be inserted in container applications supporting the ActiveX technology from Microsoft. The control can be printed together with the remaining document. Its content can be changed at will.

ActiveX controls are sometimes also referred as plug-in, element, control, component (COM component) and “Steuerelement” (in German).

ActiveX technology is supported by a broad range of applications. Most of the available development environments, e.g. Visual Basic®, Visual Studio® C/C++, C#, .NET, Borland C++ Builder, Borland Delphi, ASP, ASP.NET (and more) are supporting ActiveX. Beside that you can use ActiveX controls inside of Microsoft Office, report generators and many other applications.

A special feature of ActiveX Controls is the possibility for "data binding" – i.e. the control content can be bound to a freely selectable data field of the container application – e.g. a cell in the spreadsheet or a data field in a database.

An ActiveX object can be parameterized or controlled by the use of scripting or programming languages – or without any programming by the end-user itself.

Whenever the term ActiveX is used in this document it refers to TBarCode OCX using the ActiveX interface. However – all TEC-IT products conforming to the ActiveX specification (TBarCode OCX, TConnector, TFORMer Runtime) can be used in a similar way.

4.2 Container Application

An ActiveX control is a software component or plug-in which is embedded in a container application.

An ActiveX control comes to life with a container application only – it cannot be used as stand-alone application.

The behavior of an ActiveX control can be controlled by the following techniques:

- **Directly**
  ...by adjusting the properties of the ActiveX control by the end-user. In this case no programming knowledge is required. Just right-click the barcode object and adjust its properties.

- **Scripting**
  In Microsoft applications an ActiveX control can be controlled with VBA (Visual Basic for Applications). Sometimes JavaScript, JScript or similar scripting languages are available.

- **Programming**
  Virtually all programming languages (like Visual Basic, C, C++, C#, Pascal, Cobol, RPG, …) are able to handle ActiveX controls.

4.3 Symbology (Barcode Type)

The term symbology stands for the type of the barcode. TBarCode supports more than 70 different barcode symbologies (e.g. Code 2of5 interleaved, Code128, PDF417, Data Matrix). Each symbology specifies how to convert data (“encode” data) into the correct barcode representation.
4.4 Encoding

The term encoding stands for converting arbitrary data into a barcode. Thus – whenever you draw or print a barcode you are encoding data.

4.5 Data Binding

Data binding refers to the capability of binding TBarCode to a data field of the container application. If the content of the data field changes, the barcode content changes automatically. The remaining properties (e.g. the barcode type) remain unchanged.

Data-binding is very useful in applications like Microsoft Access or Microsoft Excel®. The barcode can be inserted into a report or spread-sheet and the content of the barcode is controlled by a table-cell (linked cell) automatically.

- Not all applications support data binding by simply adjusting a data source property. Sometimes data-binding can be achieved by scripting the ActiveX Control (our support can help you on demand).
5 Use in Common Applications

5.1 General

You can use ActiveX Controls (TBarCode, TConnector…) in all applications, which can deal with ActiveX Controls. – These are most of the commonly used standard applications and development environments.

In the next chapters we describe the handling of ActiveX Controls. In this document the TEC-IT product TBarCode OCX is used, but the steps are basically the same for all other ActiveX controls.

5.1.1 Inserting an ActiveX Control

The exact steps for each application are shown in the following chapters.

Basically the common process for inserting an ActiveX Control into an application (e.g. Microsoft Word) is as following:

- Select Insert Object or Insert ActiveX Control.
- Select TBarCode from the list.
- Insert the control – adjust the exact dimensions (with the mouse).
- Adjust the characteristics of the control (right-click ► Properties)

- Depending on the application you must be in design mode if you want to insert or edit an ActiveX Control.
- A “Toolbox” is very often used to select a control.
- In some applications you must first insert an "OLE Container", in which the TBarCode Control can be inserted.
- Make sure that Macros and ActiveX Controls are enabled (Security Settings in Microsoft Excel and Word).
6 Use in Microsoft Office Applications

6.1 Barcodes in Microsoft Word

6.1.1 Insert TBarCode into a document

In order to insert TBarCode OCX in a Microsoft Word document execute the following steps:

- Open or create your document.
- Select Insert ➤ Object… from the menu (1).
- Select TBarCode from the list of object types 2 and confirm with OK.
- To set the properties, right-click on the control 3.
- Select TBarCode Object ➤ Properties (see Figure 2 on page 12, 4).
- The property pages 5 appear. (For details on the barcode properties please refer to the product documentation).

![TBarCode OCX in Microsoft Word](image1)

Figure 1: Using TBarCode in Microsoft Word (1)
Figure 2: Using TBarCode in Microsoft Word (2)
6.1.2 Insert TBarCode for Scripting with VBA (Visual Basic for Applications)

Some requirements can be fulfilled only if you script (program) the ActiveX inside of Word. In order to access the object through an identifier (in your VBA code) you need to insert it the following way.

- Open or create your document.
- If the “Visual Basic” toolbar is not visible, right-click on the menu bar and check Visual Basic.
- If the Control Toolbox is not visible, right-click on the menu bar and check Control Toolbox.
- In the Control Toolbox (2) click on the Other Controls Icon. – A list with available controls will appear (4).
- Select the entry TBarCode. The TBarCode control will be inserted.
- Click the Properties icon in the Control Toolbox to open the property:value window.
- Now you can use TBarCode in VBA. To address the control use the object name, which is defined in the properties window.
- If you double-click the control you can enter an event-driven code (not shown).

![Figure 3: Using TBarCode in Microsoft Word with VBA](image)
6.1.3 Insert TBarCode per Code (VBA)

To create and insert a TBarCode object per VBA code into a Word document, you can use the following code:

```vba
This code shows how to insert a Code-128 symbol with the data 1234567890 at the actual cursor position.

Dim oShape As InlineShape
Dim oTBC As Object

Insert a shape containing a barcode object at the actual cursor position.

oShape.OLEFormat.Activate

For easier access link an object to the barcode object inside the shape.

Set oTBC = oShape.OLEFormat.Object

Now adjust the size of the barcode by adjusting the shape

oShape.Width = 200
oShape.Height = 100

Adjust the barcode object properties ( barcode type, encoded data...)

oTBC.Barcode = 20   'adjust Code-128
oTBC.Text = "1234567890"
```

6.1.4 Mail Merge Documents


After mail merging you can transfer the content of the selectable data field(s) to the barcode(s). To start with you need an empty barcode object in the document template.
6.2 Barcodes in Microsoft Excel

6.2.1 Insert TBarCode into a spreadsheet

To insert TBarCode in Microsoft Excel, please follow these steps:

- Open or create your document.
- Select Insert ► Object… from the menu (1).
- Select TBarCode from the list of object types (2) and confirm with OK.
- Change to design mode (triangle symbol in the Control Toolbox (3)) and select TBarCode. This – and moving the barcode object with the mouse – is only possible after the Design Mode was activated. If the Control Toolbox is not visible, open it by right-clicking the menu bar.
- To change the properties, right-click on the TBarCode object and choose TBarCode Object ► Properties (4).
- The property pages (5) appear. (For details on the barcode properties please refer to the product documentation).

Figure 4: Using TBarCode in Microsoft Excel
6.2.2 Use TBarCode with VBA (Visual Basic for Applications)

To use TBarCode within VBA, please follow these steps:

- Open or create your document.
- If the toolbar for Visual Basic is not visible, right-click on the menu bar and check Visual Basic.
- If the Control Toolbox is not visible, right-click on the menu bar and check Control Toolbox.
- In the Control Toolbox click on the More Controls icon. – A list with available controls will appear.
- Select the entry TBarCode.
- Now move the mouse cursor to the desired position and click the left mouse button. The TBarCode control will be inserted.
- Click the Properties icon in the Control Toolbox to open the property:value window.
- Now you can use TBarCode in VBA. (To address the control use the object name, which is defined in the properties window).

Figure 5: Using TBarCode in Microsoft Excel with VBA
6.2.3 Linked Cell (Data Binding)

Apart from using the control within VBA (Visual Basic for Applications) the standard property `LinkedCell` provided by Microsoft Excel is available. In the “Properties” dialog you can set the Linked Cell, which serves as source for the barcode content ("data binding").

- After `TBarCode` was inserted as described in the previous paragraph, select the `TBarCode` object.
- Click on the Properties icon in the Control Toolbox.
- Enter the linked cell in the Properties Window (here B29 is used). The barcode changes whenever you change the content of the linked cell (B29).

![Image of TBarCode in Microsoft Excel (Data Binding)](Figure 6: Using TBarCode in Microsoft Excel (Data Binding))
6.2.4 Insert TBarCode per Code (VBA)

To create and insert a TBarCode object per code, you can use the following sample:

```vba
Dim TargetCell As Variant     ' The cell, in which you want to insert the barcode
Dim CellHeight As Variant     ' Height of the target cell
Dim CellWidth As Variant      ' Width of the target cell
Dim CellTop As Variant        ' Top coordinates of the target cell
Dim CellLeft As Variant       ' Left coordinates of the target cell
Dim BCObject As OLEObject     ' The TBarCode container object

TargetCell = "C3"

' make sure that the cell height has 30 pixels minimum
If (Range(TargetCell).RowHeight < 30) Then
    Range(TargetCell).RowHeight = 30
End If

' get target cell dimensions
CellHeight = Range(TargetCell).Height
CellWidth = Range(TargetCell).Width
CellTop = Range(TargetCell).Top
CellLeft = Range(TargetCell).Left

' insert barcode object into selected cell
Set BCObject = ActiveSheet.OLEObjects.Add(ClassType:="TBARCODE6.TBarCode6.1", _
    Left:=CellLeft, _
    Top:=CellTop, _
    Width:=CellWidth, _
    Height:=CellHeight)

' set barcode properties
BCObject.Object.Text = "123456"
BCObject.Object.FontSize = 8
```

Call this code per macro or per button click, for example.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Insert BC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td><img src="image" alt="Barcode Image" /></td>
<td>123456</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Figure 7: Insert TBarCode per Code (VBA)*
6.3 Barcodes in Microsoft Access

TBarCode can be used within reports or forms. Database columns can be encoded with TBarCode directly. Besides the use of fixed values it is also possible to use formulas as data source for a barcode.

6.3.1 Insert TBarCode into a Microsoft Access Database or Report

In order to insert TBarCode into a report (or a form), please execute the following steps:

- Open or create your report (or form) in the design mode. (Window must be visible).
- From the menu select Insert ► ActiveX Control… Alternatively you can select the More Controls icon from the Toolbox. (If the Toolbox is not visible, right-click inside the toolbar-area and select Toolbox).
- Select TBarCode from the list of object types and confirm with OK.
- Move your mouse cursor to the position, where you want to insert the object (report or form) and click the left mouse button. TBarCode will be inserted.
- To set the properties right-click the TBarCode object and select TBarCode-Object ► Properties (or just Properties). The Property Window opens.

Figure 8: Using TBarCode in Microsoft Access
6.3.2 VBA (Visual Basic for Applications)

In order to be able to use TBarCode from VBA, execute the same steps as described above. Keep in mind the following points, too:

- Set the properties of the barcode object in the Access Properties Window (see Figure 8). Only in this dialog you can enter the object name as well as the control data source (list of selectable data fields).
- To display the properties, click the properties button on the menu bar or right-click the barcode object and select Properties from the context-menu.
- If you want to program the ActiveX Control in VBA you have to use the "object" interface to access the ActiveX properties and methods.

The example below gives you access to the text property of TBarCode

TBarCode61.is the name, which has been adjusted in the properties.

TBarCode61.object.text = "1234567890"

6.3.3 Data Binding

In reports and forms the barcode content can be bound to a data field. Most commonly the barcode is bound to a table column. If this is done, the barcode changes according to the actual data record displayed.

Please execute the following steps:

- Right click TBarCode OCX.
- Select Properties from the popup menu.
- Select the Data tab.
- In the data page, in the Control Source input field, you can select the data field (or the query) that serves as source for the barcode content.

This step is only possible if you have defined a data-source for the complete report. You can do this by opening the properties dialog for the entire report.

Figure 9: Using TBarCode in Microsoft Access (Data Binding)
Please ensure that the barcode type corresponds with the type of data you want to encode. If the characters cannot be encoded using the selected barcode type, an error message will be displayed instead of drawing the barcode.

You can also enter an operation as data source (e.g., a string operation like "+= field_ArticleNo + field_Price"). This might be useful for combining several data fields (ArticleNo and Price) into one barcode symbol (e.g., EAN13 P5).

### 6.4 Barcodes in Microsoft InfoPath®

Microsoft InfoPath can take advantage of barcodes by using TBarCode OCX. Especially when it comes to 2D barcodes with high data density you get a powerful solution for optimizing data flow.

- InfoPath has no built-in property dialog for ActiveX Controls, but TBarCode comes with its own property dialog - therefore you can adjust all barcode characteristics in an easy way.

#### 6.4.1 Data Binding

Controls in a form are bound to a part of the form’s data source. InfoPath stores the data of a form as XML documents. The data binding describes which part of the form is saved in the control. In TBarCode the bound data is encoded as barcode.

When you insert TBarCode, you can choose if you want to bind the barcode control to the data of a single field (called “Field (element or attribute)”) or to a part of the XML document (called “Field or Group (any data type)”).

- **Field (element or attribute):**
  Only the data of a field, which was filled in by the user, is encoded as barcode.
  For example: If the user entered “John Smith”, the encoded data for the field _singleName_ is “John Smith”.

- **Field or Group (any data type):**
  TBarCode encodes a part of the XML document. This option allows encoding of a group of fields – or even the whole form.
  For example: If the user entered “John Smith”, the encoded data for the field _singleName_ is “<invrq:singleName xmlns:invrq=http://schemas.microsoft.com/office/infopath/2003/sample/InvoiceRequest>John Smith</invrq:singleName>".
6.4.2 Prepare Microsoft InfoPath to use TBarCode OCX

Perform the following tasks to insert TBarCode into your Microsoft InfoPath document. These steps are required only if TBarCode OCX was not added to Microsoft InfoPath before:

- Open a form in Design mode.
- Open the Task Pane. Therefore select **View ► Task Pane** from the menu.
- In the Task Pane switch to **Controls**.

![Figure 10: Using TBarCode in Microsoft InfoPath (1)](image)
- Click **Add or Remove Custom Controls**…  
  You see a list of already installed custom controls. Click **Add** – the “Add Custom Control Wizard” opens.
- Locate and select **TBarCode** in the list.

![Figure 11: Using TBarCode in Microsoft InfoPath (2)](image)

- Click **Next** until the “Specify a Binding Property” page is displayed.
  - Select the binding property **Text** if you plan to store only a single field in the barcode (“Field (element or attribute)”).
  - Select the binding property **Value** if you plan to store a part of the XML document (“Field or Group (any data type)”).
- Click **Next** and select the **Enabled** entry in the “Specify and Enable or Disable Property”.
- Click **Next** and select all data types (by clicking each line) in the “Specify Data Type Options” wizard page.
  - If you have chosen the binding property **Text**, then select “Field (element or attribute)” and leave the default data type as **Text (string)**.  
  - If you have chosen the binding property **Value**, then select “Field or Group (any type)”. Then click **Finish**.
- Click **OK**.
- **TBarCode** appears in the list of available controls.
6.4.3 Insert TBarCode OCX into a Microsoft InfoPath Document

After you added **TBarCode OCX** to the available controls in Microsoft InfoPath you can add a barcode to your document by performing these steps:

- Drag the **TBarCode** control from the Task Pane to your Microsoft InfoPath form.
- After you release the mouse button a dialog named “**TBarCode Binding**” appears.
- Select the required field or group of the InfoPath document as data source for **TBarCode**. The content of this field or group will be encoded as barcode.
- Press **OK**. A barcode will be inserted into the form.

![Insert TBarCode into a Microsoft InfoPath Document](image-url)
6.4.4 Change Barcode Settings in a Microsoft InfoPath Document

- Right click the barcode.
- In the upcoming menu select **TBarCode Properties**.
- Select the desired tab (e.g. **General**).
- Select the suitable barcode type.
- Click **OK**.
- The **TBarCode** property pages offer a bunch of barcode related options. Please check out the **TBarCode** documentation for details.

![Figure 15: Changing TBarCode Settings in a Microsoft InfoPath Document](image_url)
7 Use in Development-Environments (IDE)

TEC-IT’s ActiveX Controls (e.g. TBarCode, TConnector or TFORMer) can be used in various
development environments. Instances of the object can be created, changed and printed.

- We recommend to check out the sample applications at: [http://www.tec-it.com/Download](http://www.tec-it.com/Download) ► Bar Code Software for Windows ► Samples.

Below we use our product TBarCode to describe the handling of ActiveX Controls.

7.1 Microsoft Visual Basic

7.1.1 Use TBarCode in a Visual Basic Project

In order to insert TBarCode in Visual Basic, please execute the following steps:

- Open or create your VB Project.
- Right-click the Toolbox and select Components from the menu.
- Select TBarCode Type Library from the list (Insertable Objects) and confirm with OK.
- Now you can use TBarCode as VB Control in the Toolbox.
- Set the characteristics of the barcode in the VB properties window (“Properties TBarCode61”).
- In case you prefer the TBarCode “Property Pages” (instead of the VB properties window),
right-click the object and select Properties from the popup menu. (For details please refer to
the TBarCode product documentation).

7.2 Microsoft Visual C++ 6.0

7.2.1 Use TBarCode in a MFC Application

In order to use TBarCode in MFC, please follow these steps:

- Open or create your MFC based application.
- If you create a new project, make sure, that ActiveX Controls is checked (default) on page
3 of the “MFC AppWizard”.

Figure 16: Using TBarCode in MSVC++ 6.0 (MFC Application) (1)
Select Project ➤ Add to Project ➤ Components and Controls… from the menu.
Change to the folder Registered ActiveX Controls.
Select TBarCode from the list and click Insert.
A wrapper class for TBarCode is generated automatically (2). Click OK to confirm.

- The control is thereafter selectable in the control toolbox in dialog editor (3).
- Work with the assistance of the ClassWizard if necessary.

7.2.2 Use TBarCode in a non MFC Application

For non MFC applications we recommend to use our Barcode Library TBarCode6 DLL. Samples are available on http://www.tec-it.com/Download ➤ Bar Code Software for Windows ➤ Samples.
7.3 Microsoft Visual Studio .NET 2003

7.3.1 Use TBarCode as Control

In order to use TBarCode as a control and to embed it in a form in Visual Studio .NET (e.g. in a "Windows Application" or in an "ASP .NET Web Application"), please follow these steps:

- Open or create a project.
- In the Toolbox switch to Components (1).
- Right-click inside the Toolbox and select Add/Remove Items…. A dialog appears.
- In the dialog switch to the COM Components tab (2).
- Select TBarCode (3) (check the checkbox).
- Confirm with OK (4).
- Now TBarCode OCX is added to the Toolbox (5).
- To use the control just drag it from the Toolbox (6) to your form (7).

![Figure 19: Using TBarCode in Visual Studio .Net 2003 (as Control)](image)

Microsoft Visual Studio .NET automatically generates a wrapper class for an existing ActiveX control. You can use it like any other control in .NET.
7.3.2 Use TBarCode as Library

In order to use TBarCode per code only (without embedding it as ActiveX control in a form), please follow these steps:

- Open or create a project.
- From the menu select Project ► Add Reference…
- In the upcoming dialog switch to the COM tab.
- Select TBarCode from the list and click Select.
- Confirm with OK.
- Now TBARCODELib is added to the References of your Project (and can be found in the “Solution Explorer” Window).
- The TBarCode class can now be used in your code.

![Figure 20: Using TBarCode in Visual Studio.Net 2003 (as Library)](image)

Sample usage of the TBarCode class in C#:

```csharp
TBARCODELib.TBarCodeClass obj = new TBARCODELib.TBarCodeClass();
obj.BarCode = TBARCODELib.tag_BarCode.eBC_2OF5IL;
obj.Text = "12345";
```
7.4 Microsoft Visual Studio .NET 2005

7.4.1 Use TBarCode as Control

In order to use TBarCode in Visual Studio .NET (e.g. in a “Windows Application” or in an “ASP .NET Web Application”), please follow these steps:

- Open or create a project.
- Right-click inside the Toolbox and select Choose Items…. A dialog appears.
- In the dialog switch to the COM Components tab.
- Select TBarCode (check the checkbox).
- Confirm with OK.
- Now TBarCode OCX is added to the Toolbox.
- Drag the control from the Toolbox to your form.

![Figure 21: Using TBarCode in Visual Studio .Net 2005 (as Control)](image)

Microsoft Visual Studio .NET automatically generates a wrapper class for an existing ActiveX control. You can use it like any other control in .NET.

7.4.2 Use TBarCode as Library

Use the same steps as described in 7.3.2.
7.5 Microsoft Visual FoxPro

7.5.1 Inserting TBarCode into a FoxPro-Project

In order to insert TBarCode in Visual FoxPro, please execute the following steps:

- Open or create a FoxPro Project.
- Open or create a Form.
- Select OLE Container from the Standard Toolbox.
- After drawing the outlet of the OLE container, an "Insert object" dialog is opened. Select TBarCode (Insert Control) and confirm with OK.
- The characteristics can be modified in the properties window.
- If you prefer the TBarCode property pages, right-click the control and choose Properties from the pop-up menu.
- Within the Project Manager you can add the newly created form to the project.

7.5.2 Reports & Labels

If you want to print barcodes in reports or labels, you need a table with a specific column, in which TBarCode must be stored as OLE Control (data type “general”). This data field must be initialized with the data structure of a barcode object before use. For this purpose you need an instance of a TBarCode object as template (e.g. on a form, it may be invisible).

Example code for the initialization of the data fields of a barcode object:

```
FOR n = 1 TO 10
    INSERT INTO Table1 FROM MEMVAR
    APPEND GENERAL Table1.BC CLASS TBarCode6.TBarCode6
    WITH THISFORM.[Name of the TBarCode object instance]
    .CONTROLSOURCE = "Table1.BC"
    .REFRESH
    .barcode = 20 && Barcode-Type = Code128
    .TEXT = "000070000041"
    .printdatatext = .T.
    .BACKCOLOR = 16777215
    .FORECOLOR = 0 && 16711680 -> (blue)
    .REFRESH
ENDWITH
```

After this code has been executed, an OLE Picture/ActiveX Bound Control can be inserted into a Report or Label Form. In this sample, you have to specify "Table1.BC" as data source (field). Please take a look at our FoxPro-Sample, which can be downloaded from [http://www.tec-it.com/Download](http://www.tec-it.com/Download) ➤ Bar Code Software for Windows ➤ Samples!
7.6 Microsoft Navision®

Under Navision TBarCode OCX cannot be inserted into a report directly. So you have to use the following workaround:

- Create or open a report and insert a *Picture Box*.
- Create a barcode image (per code) using the TBarCode OCX Control.
- Load the image to the *Picture Box*.

Sample code for creating a barcode image:

```cpp
COM objBC; // this is com object instantiation
objBC.Text = "121212" // set barcode data
objBC.Barcode = 20; // 20 ... use Code-128
objBC.SaveImage(DateinameBmp, 4, pxWidth, pxHeight, nXRes, nYRes);
```

For more detailed information, please take a look at our Navision Tutorial, which can be downloaded under [http://www.tec-it.com/Download](http://www.tec-it.com/Download) ► Bar Code Software for Windows ► Samples or contact our technical support.

7.7 Microsoft Office / VBA

Please refer to Use in Microsoft Office Applications.

7.8 Borland Delphi / Borland C++ Builder

7.8.1 Using TBarCode in Borland Delphi 6/7

To insert TBarCode in Delphi or Borland C++ Builder, please follow the steps described here:

- Select *Components* ► *Import ActiveX Control* from the menu.
- Select *TBarCode Type Library* from the list.
- Click the *Install* button and confirm the following popup windows with *OK*. The TBarCode Control can now be used like a standard component (Toolbar ActiveX).

For an overview over the properties and methods which are provided by the TBarCode object please refer to the product documentation. Please take a look at our samples, which can be downloaded from [http://www.tec-it.com/Download](http://www.tec-it.com/Download) ► Bar Code Software for Windows ► Samples!
7.8.2  Using TBarCode in Borland Delphi 2005 (Delphi for Win32)

To insert **TBarCode** in Delphi 2005, please follow the steps described here:

7.8.2.1  Create a package for TBarCode OCX

Choose **File ► New ► Package – Delphi for Win32** in the menu.

7.8.2.2  Saving the packages

Save the package in a separate directory. In the **Save Package As** Dialog first choose the directory (here **My Documents/Borland Studio Projects/TBarCode6**) and then give the name of the package-project (here **TBarCode6**).
7.8.2.3 Setting the Project-Options

Choose **Options** under menu **Project...**

Describe your package in the **Options Dialog** under **Description**

Now the directory must be adjusted. Choose **Directories/Conditionals** in the tree.

Under **Output directory**, choose the directory where you saved your project.

7.8.2.4 Import TBarCode Component

The next step: import **TBarCode OCX**. Choose menu **Component ▶ Import Component...**
In the Wizard, choose **Import ActiveX Control** and click **Next**.

On the next page, choose **TBARCODE OCX** and click **Next**.

On the next page, leave all default values as is. Click **Next**. The wizard will take you to the last page. Here you can choose to create a new Unit.

Click **Finish** to close the wizard. In the main window of Delphi you will find **TBARCODE6Lib_TLB**. Save this Unit in the same directory where the package-project is located.

### 7.8.2.5 Add Components to the Package

The next step: add **TBARCode OCX** to the package.

Choose menu **Project ► Add to Project...** Under **Add Unit** click **Browse** to look for **TBARCODE6Lib_TLB.PAS**

Click **OK** to add **TBARCode**.
7.8.2.6 Create and Install Package

The next step: create the package.

Choose the entry Build <ProjectName> (In our sample: Build TBarCode6) under menu Project.

The next step: install the created Package.

Choose the entry Install Packages under menu Component.

To add the created project click Add.

Choose the created package (here TBarCode6.BPL).

The chosen package (TBarCode6) is now listed in Design Packages. Click OK to end the installation.
7.8.2.7 Creating a new VCL Forms Application

Create a new VCL Forms Application.

Choose File ► New ► VCL Forms Application...

In the tool Tool Palette you can find the new category ActiveX.

TBarCode OCX was added in this category and is now ready to use.

7.9 Sybase PowerBuilder

7.9.1 TBarCode OCX in PowerBuilder DataWindow

In PowerBuilder\(^3\) the use of ActiveX Controls (like TBarCode) is not possible within so-called DataWindows, if the Control is bound to the band layer. This limitation is not caused by TBarCode.

To get dynamic “data driven” barcodes you have to use the following workaround.

Instead of placing the TBarCode OCX control inside the DataWindow, you have to insert it into the dialog Window. Thus it can be programmed for our workaround as follows.

By using this control you can create a bitmap file for each barcode, save it to disk and load it to a picture data field in the DataWindow. That has to be done for each data row/record set.

This workaround is described in detail in the following sections.

7.9.2 Insert TBarCode

To insert TBarCode into a PowerBuilder dialog window, please execute the following steps:

- Select Controls ► OLE from the menu.
- Select the page Insert Control from the Tab (PB10).
- Select TBarCode from the list.
- If desired, change the properties in the context menu (OCX Properties).

---

\(^3\) PowerBuilder version 7-10
In PowerBuilder Version 7, 8, 9 and 10 always insert TBarCode with Insert Control because Create New does not work with scripting.

7.9.3 Scripting the control within PowerBuilder 10

To access the Properties and Methods of the ActiveX Control use the following syntax:

ControlName.Object.Property

Example:

ole_1.object.text = "hello"
ole_1.object.barcode = 20  ' 20 = Code 128

The barcode type (the symbology) is set by assigning a number to the barcode property. Please refer to the documentation "Barcode Reference" (http://www.tec-it.com/download ▶ Bar Code Software for Windows ▶ Documentation) to learn more about other supported barcode symbologies.

7.9.4 Generating barcode images for the DataWindow

7.9.4.1 Generating the Barcode Images (from the Dialog Window)

You need a TBarCode OCX object in your dialog window, which can be programmed. In our sample it is called "w_barcode_printing".

Open your source code. In the section where you prepare the data for the output you have to do something like the following to save the barcode images:

//Image path and filename
ls_image = "C:\_APPS\FeeSystem\bcl.jpg"

//Saves the barcode as a JPEG
w_barcode_printing.ole_barcode.SaveImageEx (0, ls_image, 4, 100, 300, 80, 127, 127)

NOTE: Currently PowerBuilder only supports Bitmaps, JPEGs, and GIFs. You cannot use the EMF file type
With the `SetItem()` function you can finally assign the name of the generated barcode bitmap file to the column used in the DataWindow (`bc_pic`).

Using this approach will allow you to simulate the data binding inside the DataWindows.

### 7.9.4.2 Getting the Barcodes inside your DataWindow

Inside your DataWindow you have to create a column with a `STRING` data type. Here we use the name `bc_pic`. This column will contain the path and the name for each barcode bitmap file for each record. Then insert a Column control for the created data field.

In the “General” properties of the Column control check the `DISPLAY AS PICTURE` box as shown below. This will make the control show the picture and not the filename inside the control frame.

For example: If the column contains the value “c:\picture.jpg”, the control will display the picture and not the text “c:\picture.jpg”.

![PowerBuilder DataWindow “Display As Picture”](image)

Instead of using a Column control you may also insert a Computed Field control and use the `BITMAP()` function to convert the picture’s path to a graphic.

![PowerBuilder DataWindow Bitmap function](image)
7.9.4.3 Powerbuilder 10 Sample

- Check out the sample for Powerbuilder 10, which is provided with the TBarCode OCX Setup. If you use elder versions of Powerbuilder, please contact TEC-IT support for a sample: support@tec-it.com

7.10 Oracle Forms / Oracle Reports

7.10.1 Inserting into Forms

In order to insert an ActiveX Control in an Oracle Form please execute the following steps:

- Create a new form.
- In the layout editor:
  a) Draw an ActiveX control.
  b) Right-click with the mouse and select Insert Object.
  c) Select the TBarCode Control.
- Select Menu ► Program OLE Importer from the menu and import the control.

Optional:

- Create a button and add a when-button-pressed trigger on the canvas (where you can call the ActiveX control).
- Close the layout editor (This step is VERY IMPORTANT!).
- Run the form.
- Press the button to run the ActiveX.

7.10.2 Inserting into Reports

- Oracle Reports does not support ActiveX Controls.

Workaround:

The user can call an executable from within a report. For instance you can generate barcode bitmaps with the TEC-IT product Barcode Studio which can be executed in command line mode (without a visible window). Barcode Studio creates and saves a bitmap whenever it is called. Afterwards the generated bitmap can be loaded into an image inside the report.

Another possibility is to use the TFORMer product family (advanced label and form reporting tool).

Use TFORMer Designer to design your barcode label or report. Thereafter use TFORMer Runtime (or the command line tool TFPrint.exe) to generate and print the complete report. TFORMer supports direct printer output, PostScript, PDF and HTML Output directly (without any additional software).
8 Property Pages ActiveX

8.1 Invoking

In most Windows applications the content and the appearance of an ActiveX Control can be changed with a right mouse-click. After right-clicking the object a context-menu will appear. Select TBarCode-Object ► Properties from the menu to open the "Property Pages". This dialog window offers access to barcode specific settings.

➢ The Property Pages are used to change the characteristics of the Control or the OLE object – without a single line of code.

In some applications you can use the menu entry Properties (no BarCode-Object ► Properties, just Properties!), too. In this window all properties of the ActiveX (including the TBarCode properties) are listed as "property : value" pairs.

Some properties – in particular data binding (which is available in special applications like Microsoft Excel only), or OptResolution (which is supported by TBarCode OCX – but not available in the "Property Pages") – can be set up in this dialog only!
9 Contact and Support Information

TEC-IT Datenverarbeitung GmbH

Address: Wagnerstr. 6
AT-4400 Steyr
Austria/Europe

Phone: +43 / (0) 7252 / 72 72 0
Fax: +43 / (0) 7252 / 72 72 0 – 77
Email: mailto:support@tec-it.com
Web: http://www.tec-it.com

AIX is a registered trademark of IBM Corporation.
HTML, DHTML, XML, XHTML are trademarks or registered trademarks of W3C, World Wide Web Consortium, Laboratory for Computer Science NE-43-358, Massachusetts Institute of Technology, 545 Technology Square, Cambridge, MA 02139.
JAVA® is a registered trademark of Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA 94303 USA.
JAVASCRIPT® is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.
Microsoft®, Windows®, Microsoft Word®, Microsoft Excel® are registered trademarks of Microsoft Corporation.
Navision is a registered trademark of Microsoft Business Solutions ApS in the United States and/or other countries.
Oracle® is a registered trademark of Oracle Corporation.
PCL® is a registered trademark of the Hewlett-Packard Company.
PostScript is a registered trademark of Adobe Systems Inc.
SAP, SAP Logo, R/2, R/3, ABAP, SAPscript are trademarks or registered trademarks of SAP AG in Germany (and in several other countries).

All other products mentioned are trademarks or registered trademarks of their respective companies. If any trademark on our web site or in this document is not marked as trademark (or registered trademark), we ask you to send us a short message (mailto:office@tec-it.com).
# Document History

<table>
<thead>
<tr>
<th>Date</th>
<th>Author</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>06-Oct-2005</td>
<td>GuenterK</td>
<td></td>
</tr>
<tr>
<td>07-Mar-2006</td>
<td>StephanG</td>
<td>New Document Layout</td>
</tr>
</tbody>
</table>