

Table of Contents

Foreword	0
Part I Introduction	3
Part II Installation	3
1 Trial Version.....	3
2 Full Version.....	3
Part III How to Distribute It	3
Part IV Office 2007	3
1 Word	3
Install Template File	3
Create Single Barcode	4
Create Multiple Barcodes	5
Mail Merge	6
2 Excel	8
Change Settings	8
Create Multiple Barcodes	9
Part V Office 2000 & 2003	10
1 Word.....	10
Install Template File	10
Create Single Barcode	12
Create Multiple Barcodes	13
Mail Merge	14
2 Excel.....	15
Change Settings	15
Create Multiple Barcodes	16
Part VI Reference Guide	18
1 BCBearerBar Function.....	18
2 BCCodaBarInfo Function.....	18
3 BCCodeOne Function.....	19
4 BCConfigure Function.....	19
5 BCCopyToClipboard Function.....	20
6 BCFontInfo Function.....	20
7 BCGetActualSize Function.....	21
8 BCMicroPDF417 Function.....	21
9 BCMicroQRCode Function.....	22
10 BCRender Function.....	22
11 BCSaveAsBMP Function.....	22

12	BCSaveAsWMF Function.....	23
13	BCSetBackColor Function.....	24
14	BCSetBarColor Function.....	24
15	BCSetDefault Function.....	24
16	BCSetMessage Function.....	24
17	BCSetSize Function.....	25
18	BCSupplementInfo Function.....	26
19	BCTextInfo Function.....	26
20	BCUPCESystem Function.....	27
	Part VII Barcode Types	27
	Part VIII Convert Size from CMs to Pixels	29
	Part IX License	30
	Index	0

1 Introduction

MW6 Barcode Win32 DLL supports over 90 symbologies including Code 39, Code 128, GS1-128, EAN 13, EAN 8, UPC-A, UPC-E, Royal Mail 4 State, USPS OneCode, Deutsche Post Identcode, Deutsche Post Leitcode, Japan Postal Code, Micro PDF417, Micro QRCode, CodaBlock-F, Code 16K and Code 49, please check out this page for complete list of featured barcodes.

It is workable in Word, Access, Excel, Visual Basic, Visual C++, Visual FoxPro, Delphi, C++ Builder, VB.NET and C#.NET, you can save the barcode as BMP file or WMF file for high quality printing.

2 Installation

2.1 Trial Version

1. UnZip MW6BarcodeWin32.ZIP, run the setup.exe to install Barcode Win32 DLL.
2. The trial version Barcode Win32 DLL adds "MW6 Demo" at the top of barcode.

2.2 Full Version

1. Uninstall the trial version Barcode Win32 DLL if applicable.
2. UnZip full version Barcode Win32 DLL .zip file and run the setup.exe to install the full version Barcode Win32 DLL.

3 How to Distribute It

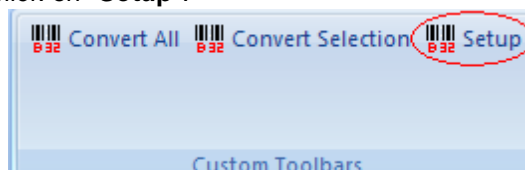
If you want to redistribute the Barcode Win32 DLL as part of your application, on the target machine, simply put **BarcodeWin32.dll** into the windows 32-bit system folder (e.g. "c:\windows\system32" or "c:\winnt\system32") for 32-bit Windows OS, or the SysWow64 folder (e.g. "c:\windows\SysWow64") for 64-bit Windows OS.

4 Office 2007

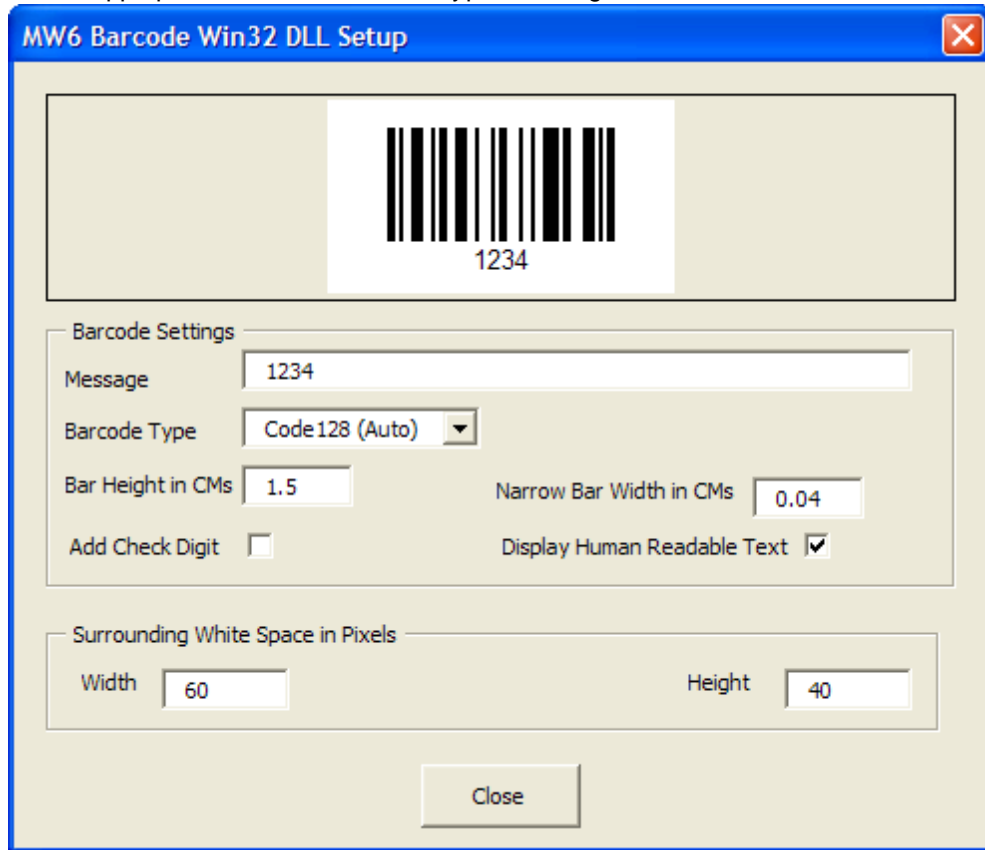
4.1 Word

4.1.1 Install Template File

1. Locate Microsoft Word Startup folder, which usually is "C:\Documents and Settings\\Application Data\Microsoft\Word\STARTUP".
2. Copy MW6_Barcode_Win32.dotm to this folder.
3. Click on "**Add-Ins**", then click on "**Setup**".

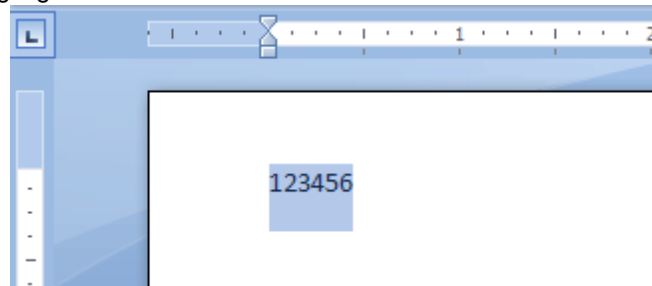


4. Choose a few appropriate values for barcode type, bar height, narrow bar width, etc.

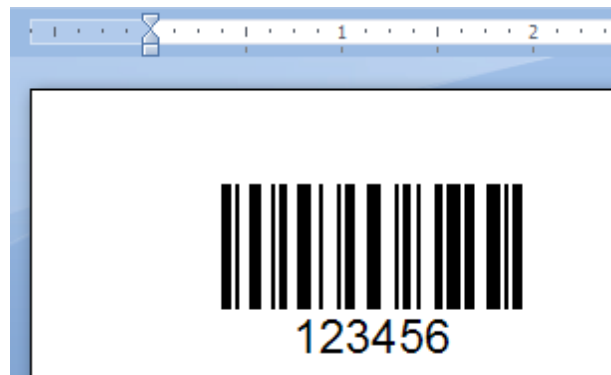


4.1.2 Create Single Barcode

1. Enter a string and highlight it.

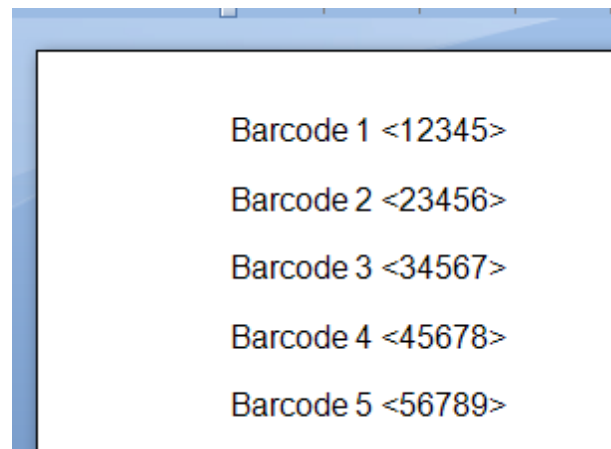


2. Click on "**Add-Ins**", then click on "**Convert Selection**" to create a barcode.
-

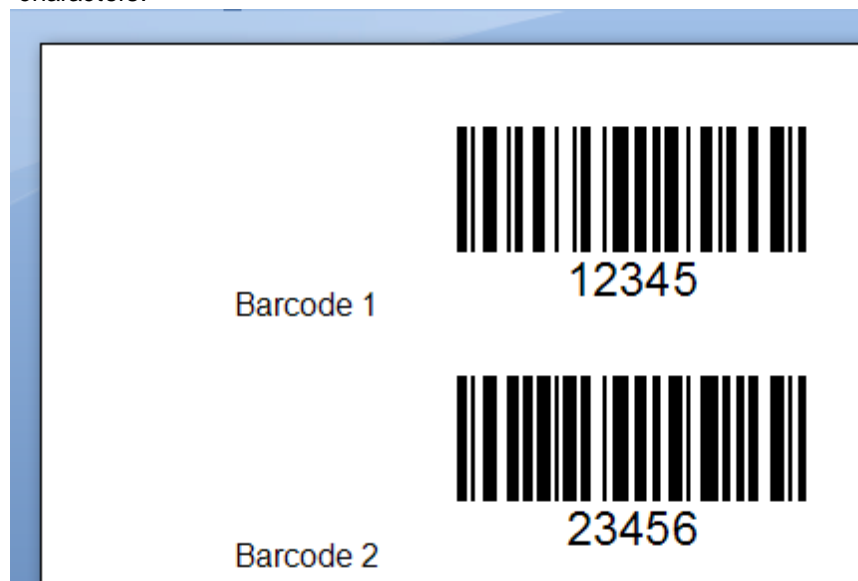


4.1.3 Create Multiple Barcodes

1. Enter a few strings, surround the strings which will be converted to the barcodes with the "<" and ">" characters.

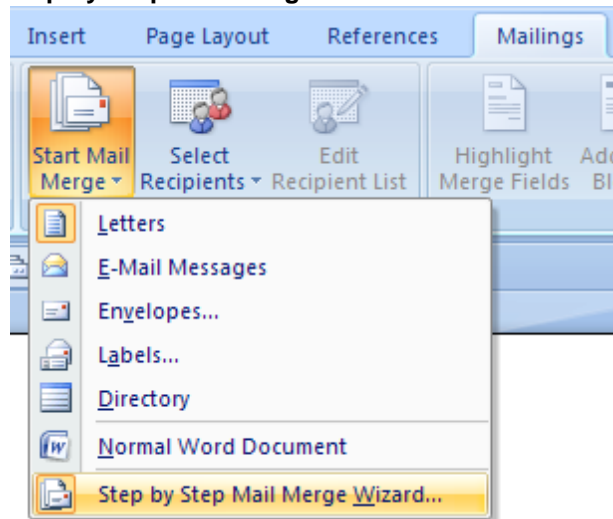


2. Click on "**Add-Ins**", then click on "**Convert All**" to create the barcodes for the strings surrounded with the "<" and ">" characters.

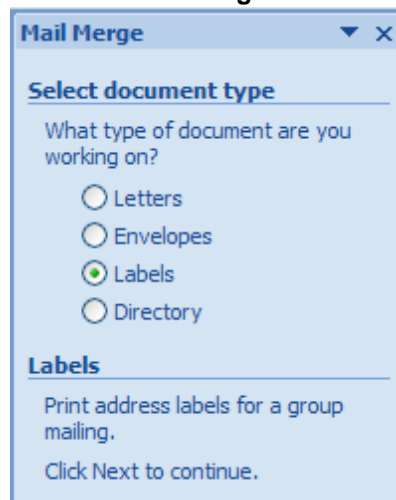


4.1.4 Mail Merge

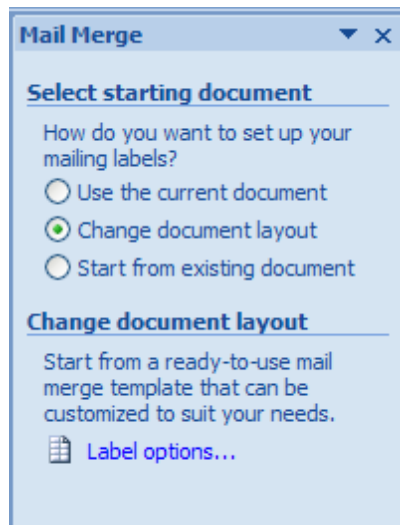
1. Click on "**Mailings**", then click on "**Start Mail Merge**". A drop-down list appears as shown below, select the last option "**Step by Step Mail Merge Wizard**".



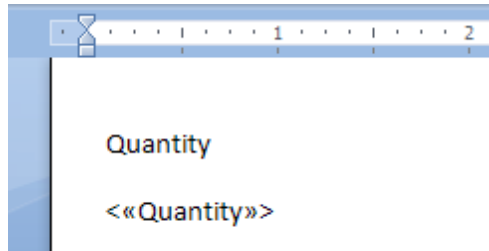
2. Select a document type and click on "**Next: Starting document**".



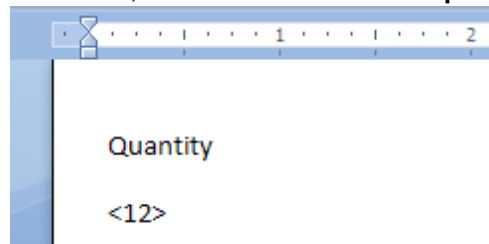
3. Click on "**Change document layout**", then choose an appropriate option and click "**Ok**", click on "**Next: Select recipients**".
-



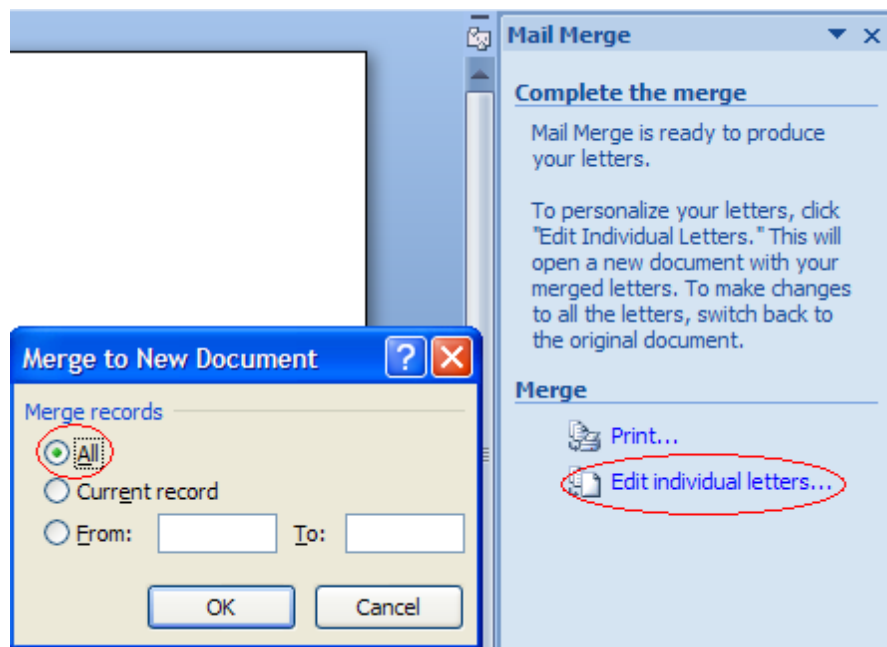
4. Select **"Use an existing list"** and click on **"Browser"** link, choose a database as an existing list, click **"Next: Arrange your labels"**.
5. Surround the string which will be converted to a barcode with the "<" and ">" characters.



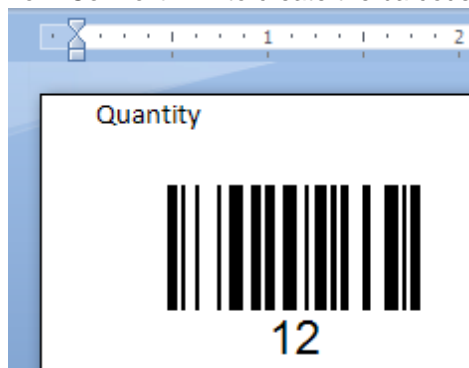
6. Click on **"Next: Preview your letters"**, then click on **"Next: Complete the merge"**.



7. Click on **"Edit individual letters"**, this opens up **"Merge to New Document"** dialog, click on **"All"** and then click on **"OK"** button.



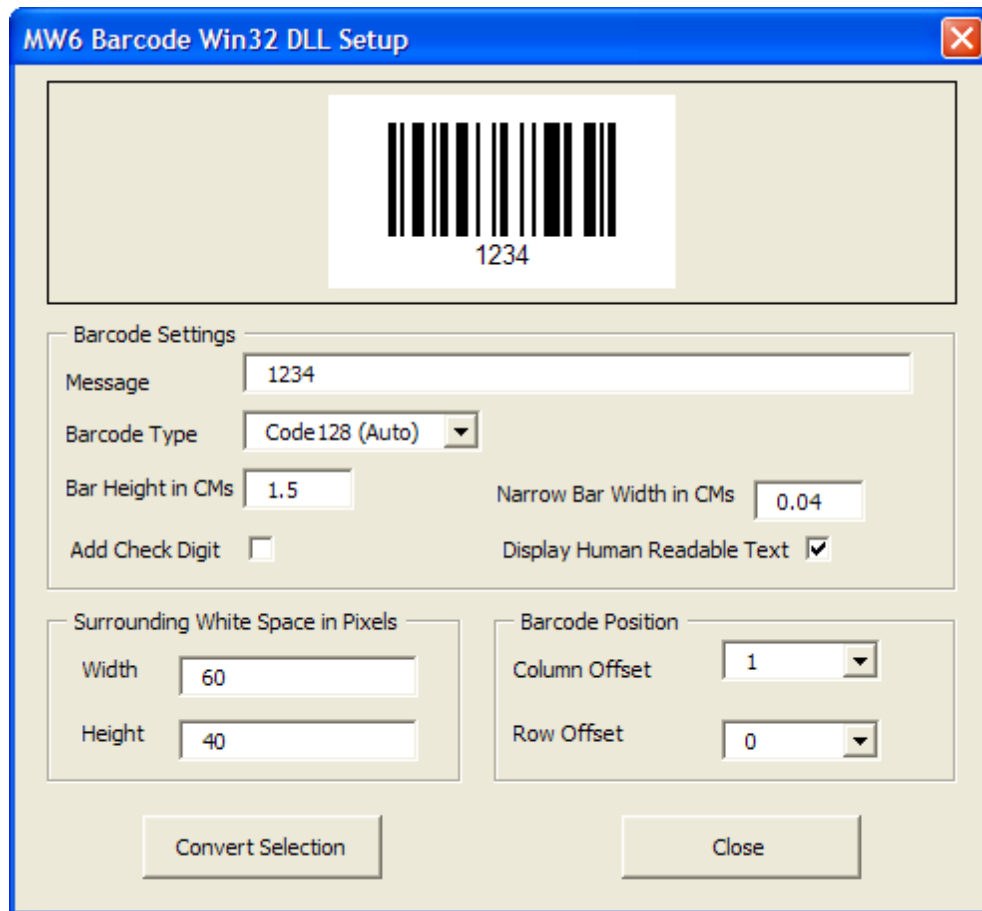
8. Click on "Add-Ins", then click on "Convert All" to create the barcodes.



4.2 Excel

4.2.1 Change Settings

1. In Excel, open MW6_Barcode_Win32.xlsm.
2. If you see "Security Warning, Macros have been disabled", click on "Options" to open "Microsoft Office Security Options" dialog, toggle on "Enable this content" check box.
3. Click on "Developer" > "Macros", select "MW6_Barcode_Win32_Setup" and click "Run".



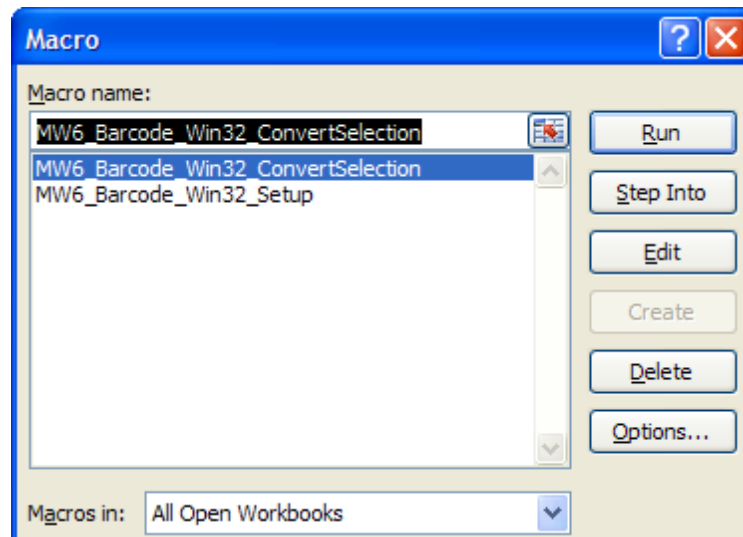
4. Choose a few appropriate values for barcode type, bar height, narrow bar width, etc., "**Column Offset**" and "**Row Offset**" are used to specify the barcode position relative to the position of a cell which contains the regular string.

4.2.2 Create Multiple Barcodes

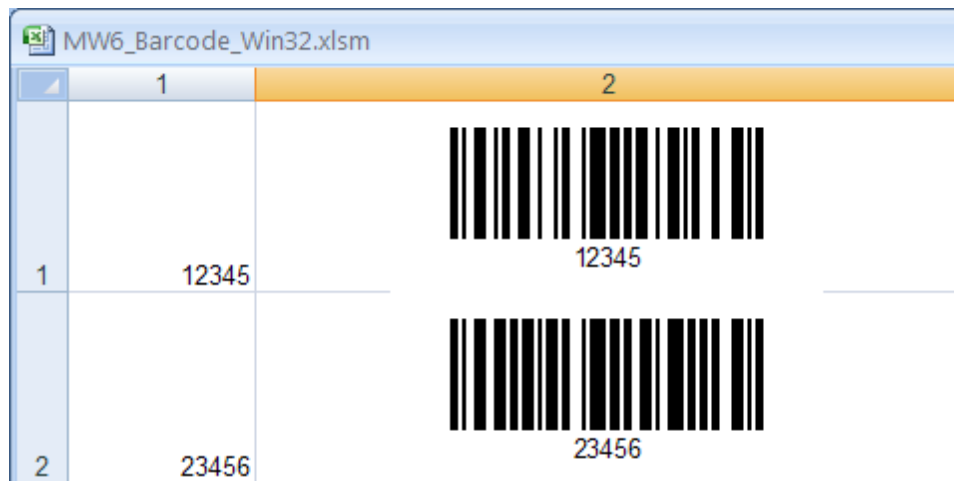
1. Select a few cells.

	1	2
1	12345	
2	23456	
3	34567	
4	45678	
5	56789	
6		
7		

2. Click on "**Developer**" > "**Macros**", select "**MW6_Barcode_Win32_ConvertSelection**".



3. Click on "Run" to create the barcodes for the selected cells.



5 Office 2000 & 2003

5.1 Word

5.1.1 Install Template File

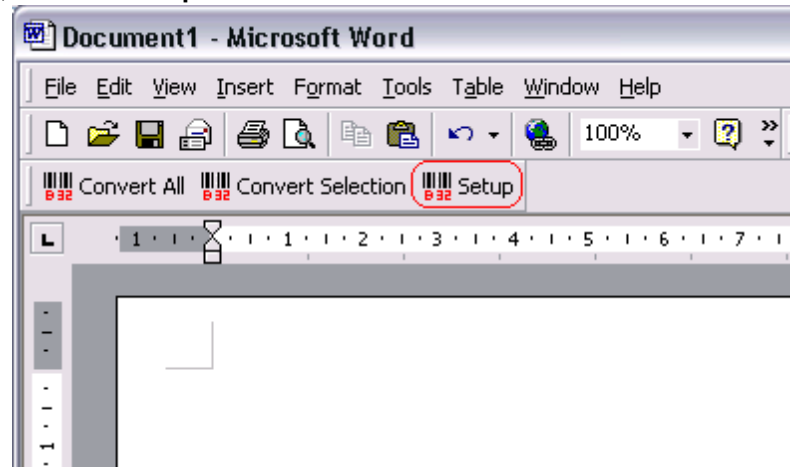
1. Locate the Word Startup folder, the Startup folder can be found in the following locations:

OS	Location
Windows 2000/XP	"C:\Documents and Settings\ <user name="">\Application Data\Microsoft\Word\Startup"</user>
Windows NT4	"C:\Winnt\Profiles\ <user name="">\Application Data\Microsoft\Word\Startup"</user>

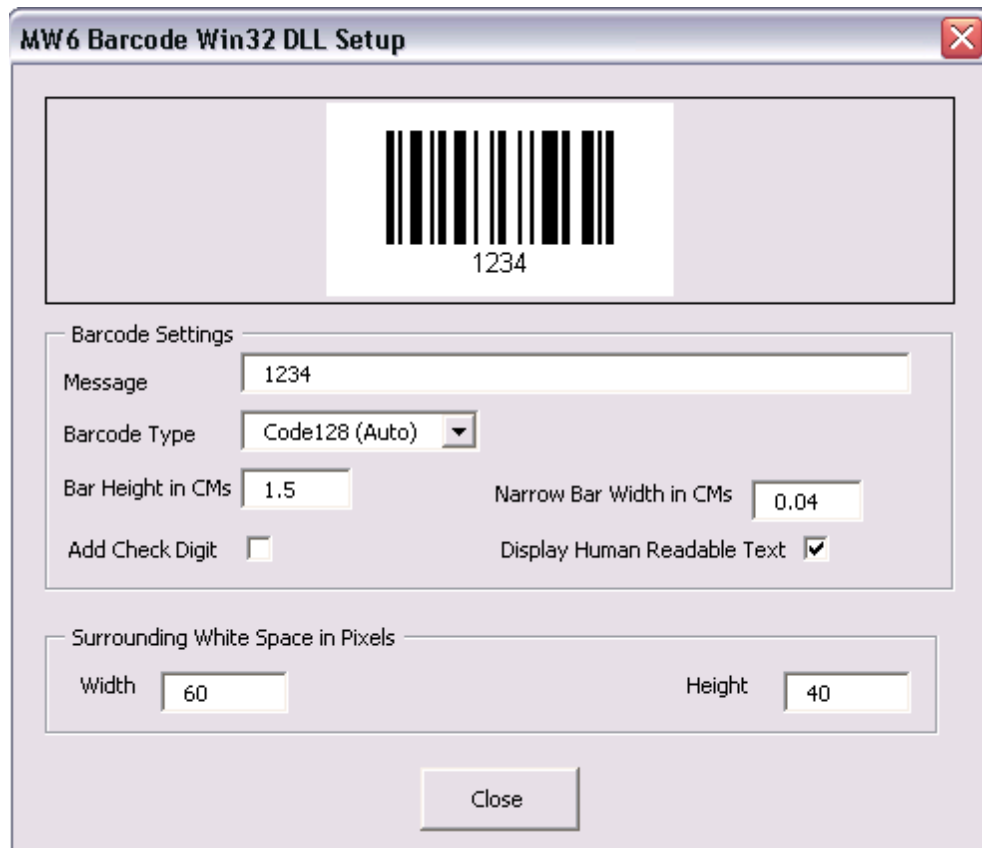
Windows 95, 98, ME	Office XP: "C:\Program Files\Microsoft Office\Office10\Startup" Office 2000/97: "C:\Program Files\Microsoft Office\Office\Startup"
--------------------	---

2. Copy MW6_Barcode_Win32.dot, which usually is in the folder "c:\Program Files\MW6 Win32 DLL \Barcode", to the Word Startup folder.

3. Open up Word, click on "**Setup**".

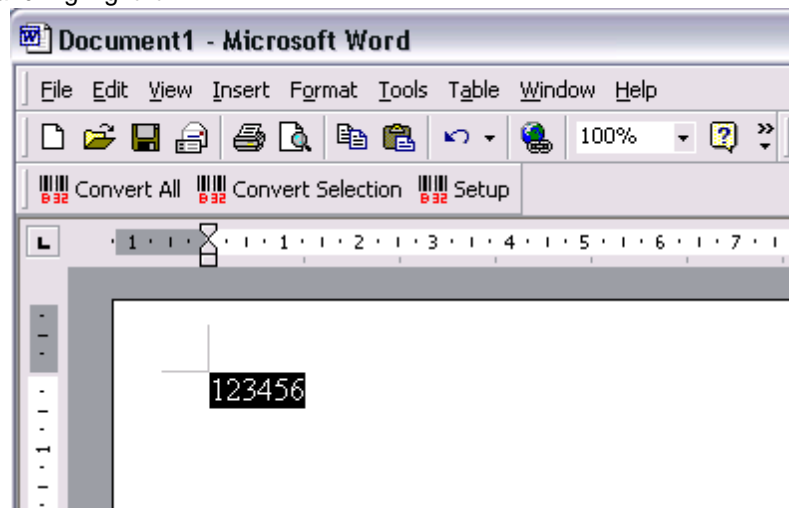


4. Choose a few appropriate values for barcode type, bar height, narrow bar width, etc.

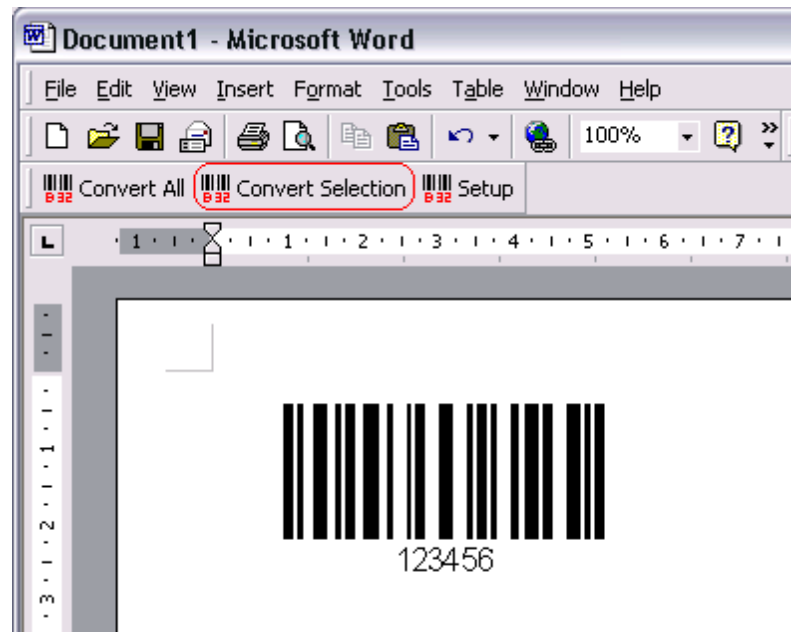


5.1.2 Create Single Barcode

1. Enter a string and highlight it.

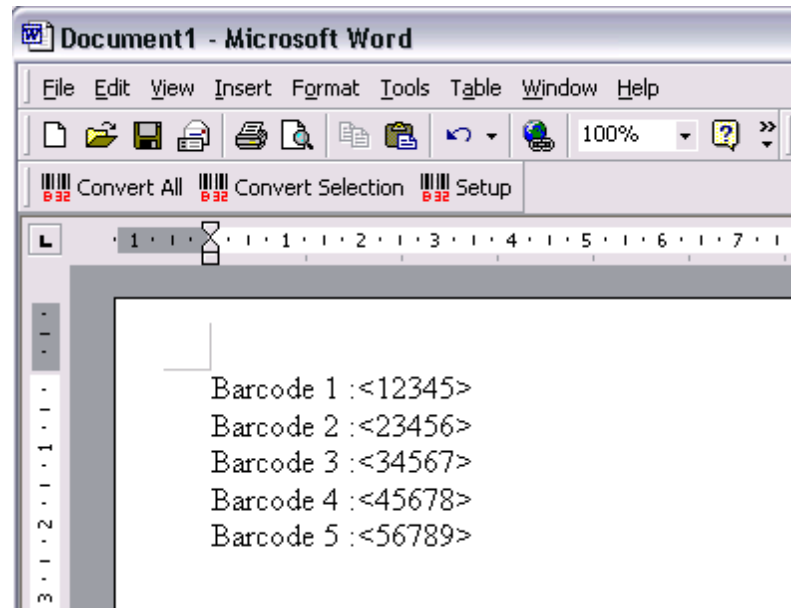


2. Click on "Convert Selection" to create a barcode.

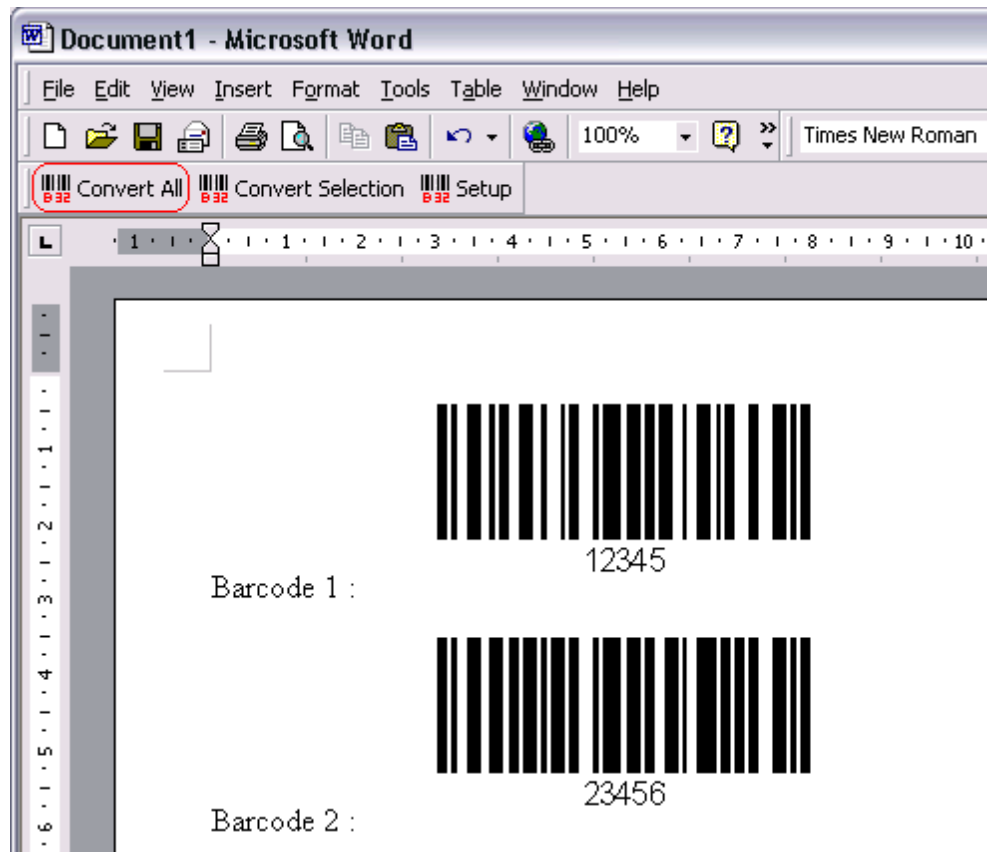


5.1.3 Create Multiple Barcodes

1. Enter a few strings, surround the strings which will be converted to the barcodes with the "<" and ">" characters.

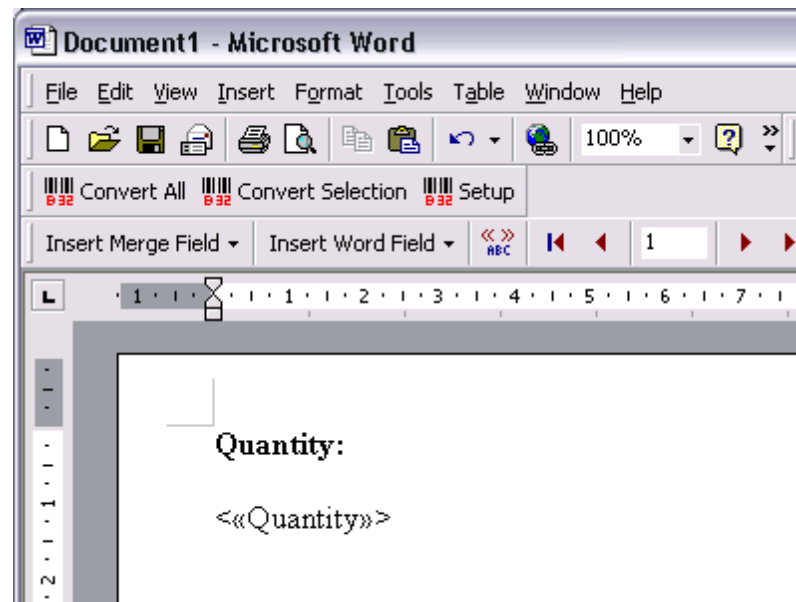


2. Click on "**Convert All**" to create the barcodes for the strings surrounded with the "<" and ">" characters.



5.1.4 Mail Merge

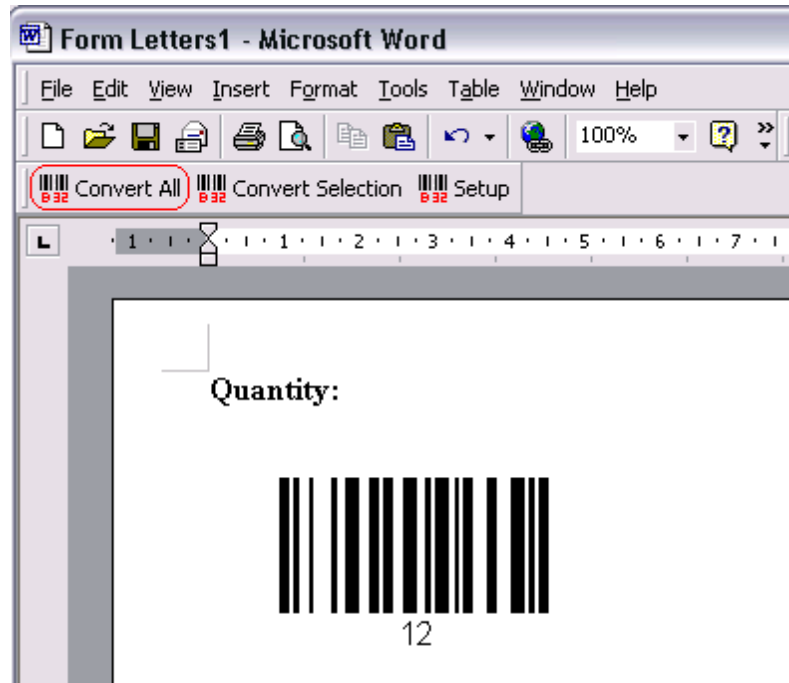
1. In Mail Merge, surround the strings which will be converted to the barcodes with the "<" and ">" characters.



2. Click on "Merge ..."



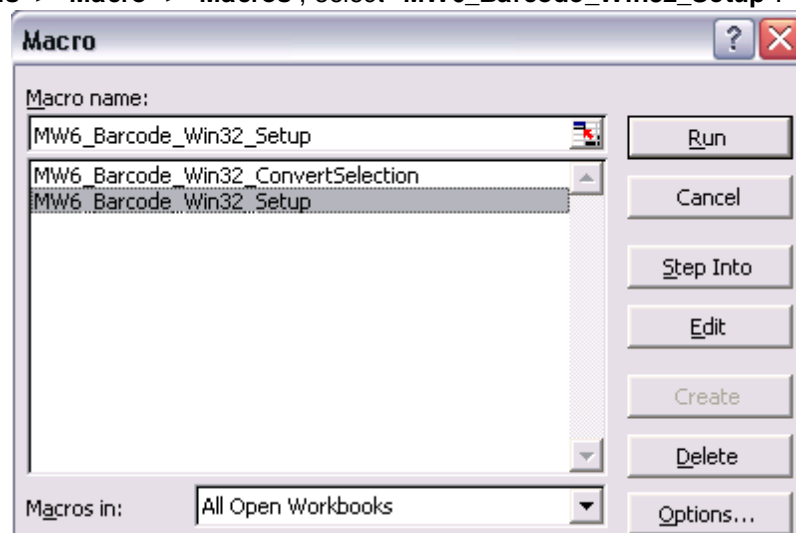
3. Click on "**Convert All**" to create the barcodes for the strings surrounded with the "<" and ">" characters.



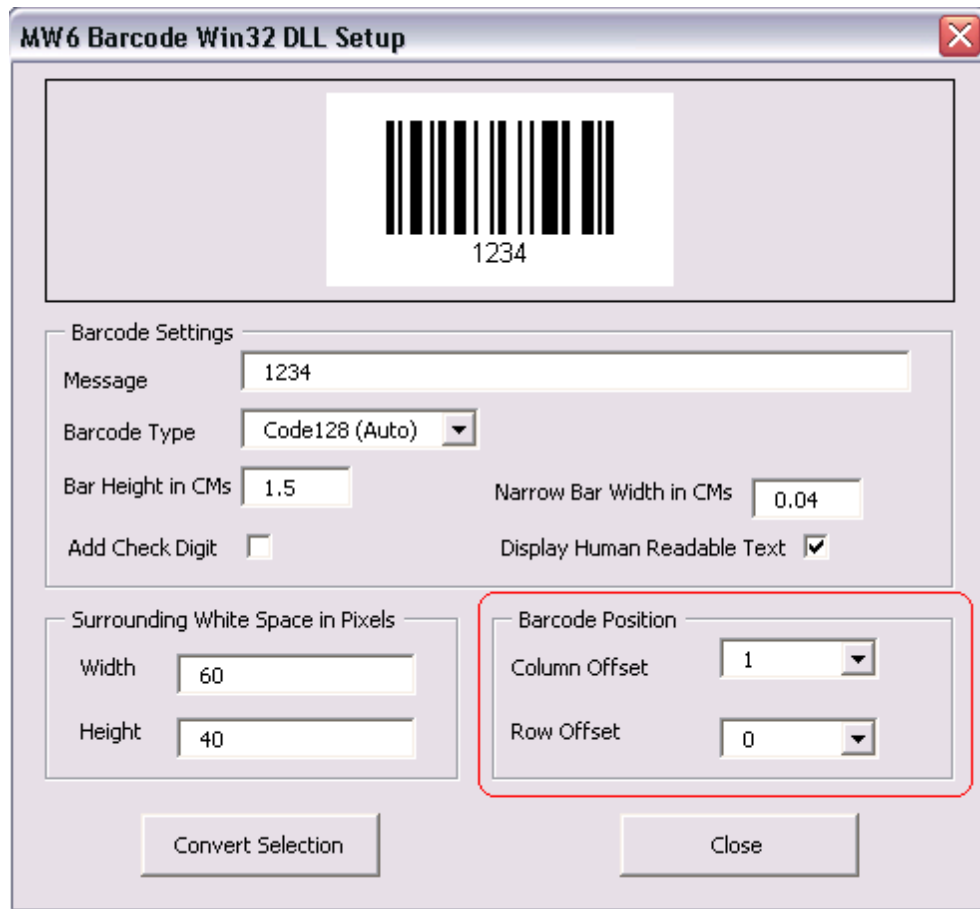
5.2 Excel

5.2.1 Change Settings

1. In Excel, open MW6_Barcode_Win32.XLS.
2. Click on "**Tools**" > "**Macro**" > "**Macros**", select "**MW6_Barcode_Win32_Setup**".



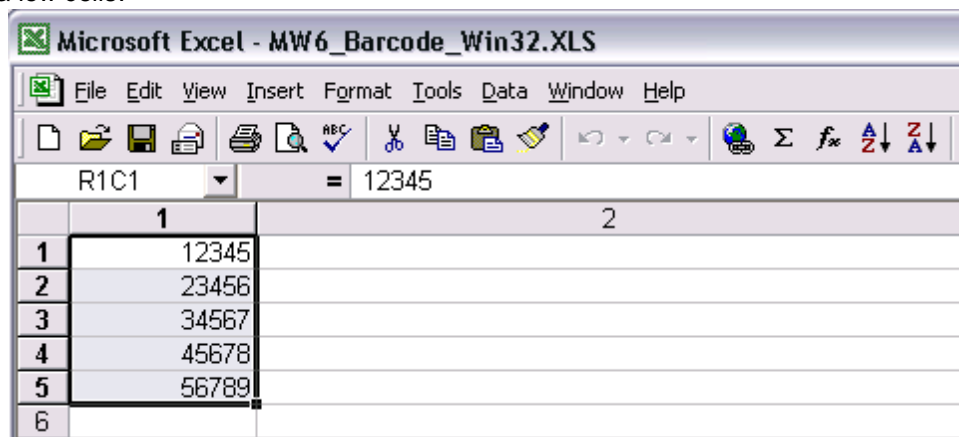
3. Click on "**Run**".



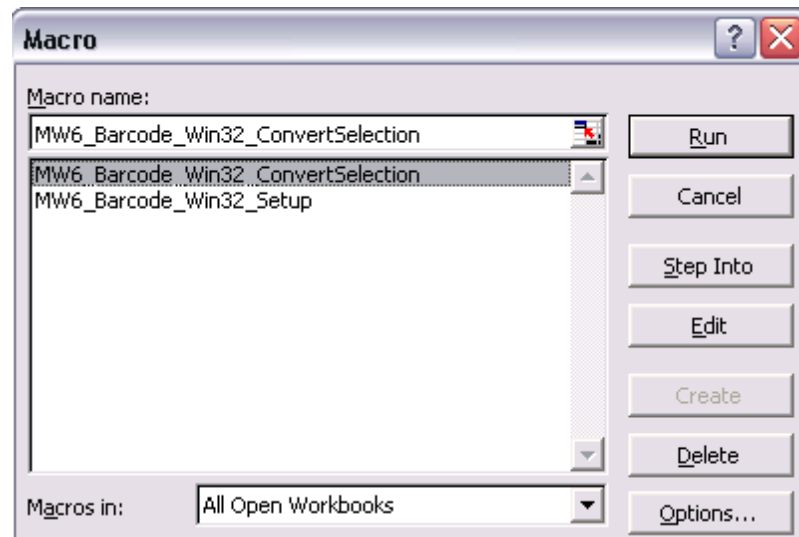
4. Choose a few appropriate values for barcode type, bar height, narrow bar width, etc., "**Column Offset**" and "**Row Offset**" are used to specify the barcode position relative to the position of a cell which contains the regular string.

5.2.2 Create Multiple Barcodes

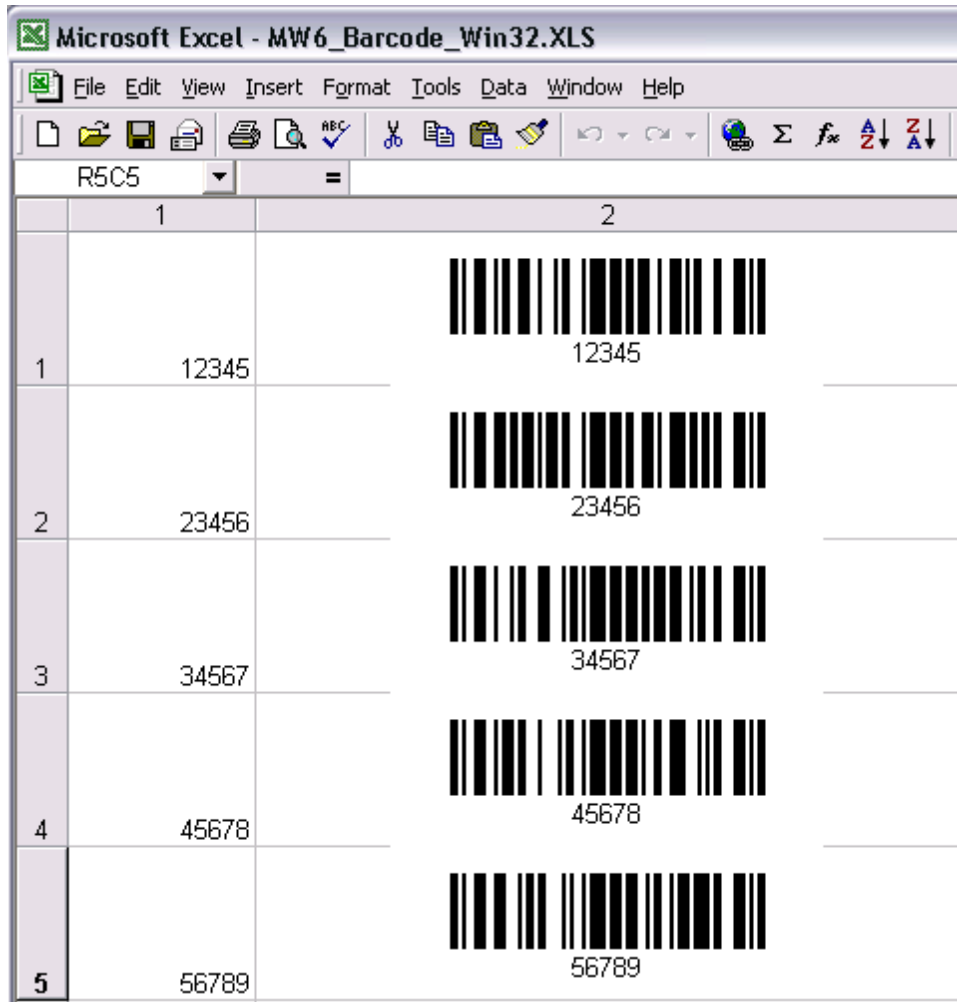
1. Select a few cells.



2. Click on "**Tools**" > "**Macro**" > "**Macros**", select "**MW6_Barcode_Win32_ConvertSelection**".



3. Click on "Run" to create the barcodes for the selected cells.



6 Reference Guide

6.1 BCBearerBar Function

Specifies the style of bearer bar for a few kinds of barcodes.

```
void BCBearerBar(WORD style);
```

Parameters

style

Specifies the style of bearer bar, this parameter can be one of the following values:

Value	Description
0	No bearer bar
1	Horizontal bearer bars only
2	A bearer bar box around the barcode

Remarks

Please check out this page to find out which format barcode is allowed to carry the bearer bars.

6.2 BCCodaBarInfo Function

Specifies the start character and the end character of CodaBar barcode .

```
void BCCodaBarInfo(WORD StartChar, WORD EndChar);
```

Parameters

StartChar

Specifies the start character of CodaBar, this parameter can be one of the following values:

Value	Description
0	Start character 'A'
1	Start character 'B'
2	Start character 'C'
3	Start character 'D'

EndChar

Specifies the end character of CodaBar, this parameter can be one of the following values:

Value	Description
0	End character 'A'
1	End character 'B'
2	End character 'C'
3	End character 'D'

Remarks

If you don't use CodaBar barcode, ignore this function.

6.3 BCCodeOne Function

Specifies the version of 2D Code One barcode.

```
void BCCodeOne(WORD version);
```

Parameters

version

Specifies the version of Code One, this parameter can be one of the following values:

Value	Size
1	16 X 18
2	22 X 22
3	28 X 32
4	40 X 42
5	52 X 54
6	70 X 76
7	104 X 98
8	148 X 134
9	8 X varied width
10	16 X varied width

6.4 BCConfigure Function

Specifies the barcode parameters such as type, bar height, narrow bar width, ratio of wide bar to narrow bar, orientation and border width.

```
void BCConfigure(WORD SymbologyType,  
                double BarHeight,  
                double NarrowBarWidth,  
                double Wide2NarrowRatio,  
                WORD Orientation,  
                double BorderWidth);
```

Parameters

SymbologyType

Specifies the barcode code type, the values of all types are listed here.

BarHeight

Specifies the bar height in centimeters, please refer to this note for more information.

NarrowBarWidth

Specifies the narrow bar width in centimeters, please refer to this note for more information.

Wide2NarrowRatio

Specifies the ratio of the wide bar to the narrow bar, typically this parameter value should be between 2 and 3.

Orientation

Specifies the orientation of the barcode, this parameter can be one of the following values:

Value	Description
0	0 degree
1	90 degrees
2	180 degrees
3	270 degrees

BorderWidth

Specifies the border width in centimeters, a valid value must be between 0 and 1.

6.5 BCCopyToClipboard Function

Copies the barcode WMF format image into the system clipboard.

```
BOOL BCCopyToClipboard();
```

Return Value

If the function succeeds, the return value is a nonzero value, otherwise the return value is zero.

Remarks

Before you call this function, use `BCGetActualSize()` function to obtain the actual size of barcode and use `BCSetSize()` function to set image size by adding surrounding white space around the barcode.

See Also

`BCGetActualSize()` Function | `BCSetSize()` Function

6.6 BCFontInfo Function

Specifies the font characteristics of text in the barcode.

```
void BCFontInfo(LPCTSTR FontName, WORD FontSize, BOOL Bold, BOOL Italic);
```

Parameters

FontName

A string that contains the font name.

FontSize

Indicates the font size.

Bold

Indicates whether a bold style is applied to the font or not.

Italic

Indicates whether an italic style is applied to the font or not.

6.7 BCGetActualSize Function

Gets the actual size of barcode which is rendered onto either computer screen or other devices such as printers.

```
void BCGetActualSize(BOOL ScreenIsTarget,  
                    DWORD TargetHDC,  
                    DWORD *ActualWidth,  
                    DWORD *ActualHeight);
```

Parameters

ScreenIsTarget

Indicates whether barcode is rendered onto computer screen or not.

TargetHDC

Device context on which to render the barcode, if the parameter *ScreenIsTarget* is set to TRUE, set this parameter to NULL.

ActualWidth

A pointer to the variable that receives the width of barcode (in pixels).

ActualHeight

A pointer to the variable that receives the height of barcode (in pixels).

6.8 BCMicroPDF417 Function

Specifies the number of columns for 2D Micro PDF417 barcode.

```
void BCMicroPDF417(WORD columns);
```

Parameters

columns

Specifies the number of columns, this parameter can be one of the following values:

Value	Description
1	1 column
2	2 columns
3	3 columns
4	4 columns

6.9 BCMicroQRCode Function

Specifies the version and error correction level for 2D Micro QRCode barcode.

```
void BCMicroQRcode(WORD version, WORD level);
```

Parameters

version

Specifies the version of Micro QRCode, this parameter can be one of the following values:

Value	Description
1	Version M1 with the size 11 X 11
2	Version M2 with the size 13 X 13
3	Version M3 with the size 15 X 15
4	Version M4 with the size 17 X 17

level

Specifies the error correction level of Micro QRCode, this parameter can be one of the following values:

Value	Description
1	L (applicable to version M2, M3 and M4)
2	M (applicable to version M2, M3 and M4)
3	Q (applicable to version M4 only)

6.10 BCRender Function

Renders the barcode onto the device such as computer screen or printers.

```
void BCRender(DWORD hDC, WORD x, WORD y);
```

Parameters

hDC

Device context on which to render the barcode.

x

The x coordinate, in pixels, of the top left corner of barcode .

y

The y coordinate, in pixels, of the top left corner of barcode.

6.11 BCSaveAsBMP Function

Saves the barcode image as a BMP file.

```
BOOL BCSaveAsBMP(LPCTSTR FileName);
```

Parameters

FileName

A string that contains the name of the file to which to save BMP format barcode image.

Return Value

If the function succeeds, the return value is a nonzero value, otherwise the return value is zero.

Remarks

Before you call this function, use `BCGetActualSize()` function to obtain the actual size of barcode and use `BCSetSize()` function to set image size by adding surrounding white space around the barcode.

See Also

`BCGetActualSize()` Function | `BCSetSize()` Function

6.12 BCSaveAsWMF Function

Saves the barcode image as a WMF file.

```
BOOL BCSaveAsWMF(LPCTSTR FileName);
```

Parameters

FileName

A string that contains the name of the file to which to save WMF format barcode image.

Return Value

If the function succeeds, the return value is a nonzero value, otherwise the return value is zero.

Remarks

Before you call this function, use `BCGetActualSize()` function to obtain the actual size of barcode and use `BCSetSize()` function to set image size by adding surrounding white space around the barcode.

See Also

`BCGetActualSize()` Function | `BCSetSize()` Function

6.13 BCSetBackColor Function

Specifies the RGB triplet of background color.

```
void BCSetBackColor(WORD red, WORD green, WORD blue);
```

Parameters

red

Specifies the value of red component for a RGB triplet, the valid value should be between 0 and 255.

green

Specifies the value of green component for a RGB triplet, the valid value should be between 0 and 255.

blue

Specifies the value of blue component for a RGB triplet, the valid value should be between 0 and 255.

6.14 BCSetBarColor Function

Specifies the RGB triplet of bar and text color.

```
void BCSetBarColor(WORD red, WORD green, WORD blue);
```

Parameters

red

Specifies the value of red component for a RGB triplet, the valid value should be between 0 and 255.

green

Specifies the value of green component for a RGB triplet, the valid value should be between 0 and 255.

blue

Specifies the value of blue component for a RGB triplet, the valid value should be between 0 and 255.

6.15 BCSetDefault Function

Initialize the barcode parameters with the default values.

```
void BCSetDefault();
```

6.16 BCSetMessage Function

Specifies the message to encode using selected barcode format and appropriate parameters.

```
void BCSetMessage(LPCTSTR Message);
```

Parameters*Message*

A string that contains the message to encode using selected barcode format and appropriate parameters.

Remarks

If the SymbologyType is set to UCC/EAN128 (GS1-128), you could use a "FNC1" string to indicate the end of a varied-length data field. For example, pass "(01)12345678901234(10)12345FNC1(11)080101" to the "Message" parameter, "FNC1" is used to indicate the end of the data field after the Application Identifier (AI) #10, since AI #10 allows the corresponding data field to have 1-20 alphanumeric characters.

**See Also**

BCBasicInfo() Function

6.17 BCsetSize Function

Sets the size of image containing the created barcode.

```
void BCsetSize(WORD Width, WORD Height);
```

Parameters*Width*

The width, in pixels, of the image.

Height

The height, in pixels, of the image.

Remarks

First call BCGetActualSize() function to obtain the actual size of the barcode, then use this function to set image size by adding surrounding white space around the barcode.

See Also

BCGetActualSize() Function

6.18 BCSupplementInfo Function

Specifies the barcode supplement parameters.

```
void BCSupplementInfo(WORD SupplementType, LPCTSTR Supplement, double Gap);
```

Parameters

SupplementType

Specifies the supplement type, this parameter can be one of the following values:

Value	Description
0	None
1	Supplement 2
2	Supplement 5

Supplement

Specifies the supplement string to encode with UPC or EAN.

Gap

Specifies the distance, in centimeters, between the normal barcode and the supplement section.

Remarks

If you don't need a supplement for the barcode, ignore this function.

6.19 BCTextInfo Function

Indicates whether to display human readable text, add check digit and display check digit in human readable text.

```
void BCTextInfo(BOOL ShowText, BOOL CheckDigit, BOOL CheckDigitToText);
```

Parameters

ShowText

Sets a boolean flag indicating whether the human readable text should be displayed or not.

CheckDigit

Sets a boolean flag indicating whether the check digit is required or not.

CheckDigitToText

Sets a boolean flag indicating whether the check digit should be displayed in the barcode human readable text or not.

6.20 BCUPCESystem Function

Specifies the encoding system of the UPC-E barcode.

```
void BCUPCESystem(WORD SystemType);
```

Parameters

SystemType

This parameter can be one of the following values:

Value	Description
0	System 0
1	System 1

Remarks

If you don't use UPC-E barcode, ignore this function.

7 Barcode Types

This table lists all barcode types supported by the Barcode Win32 DLL:

Symbology Type Value	Barcode Description	Allow Barer Bars?	Allow Supplement 2 or 5?	Sample Barcode String
1D Barcodes				
1	Channel Code			
2	Codabar			
3	Code 11			
4	Code 128			1234ABCD+ /
5	Code 128 (Set A)			
6	Code 128 (Set B)			
7	Code 128 (Set C)			
8	Code 32 or Italian Pharmacode			
9	Code 39			1234ABCD
10	Code 39 Extended			
11	Code 93			
12	Data Logic 2/5	Yes		
13	EAN128/UCC (GS1-128)			(21)95FNC1(11)090101
14	EAN 13		Yes	123456789012
15	EAN 8		Yes	1234567
16	EAN Velocity		Yes	
17	Flattermarken			
18	GS1 Databar-14			1234567890123
19	GS1 DataBar Expanded			
20	GS1 DataBar Expanded Stacked			

21	GS1 Databar Limited			
22	GS1 Databar Stacked			
23	GS1 DataBar Stacked Omnidirectional			
24	GS1 Databar Truncated			
25	GS1 Databar-14 Composite			
26	GS1 DataBar Expanded Composite			(01)1234567890123
27	GS1 DataBar Expanded Stacked Composite			
28	GS1 Databar Limited Composite			
29	GS1 Databar Stacked Composite			
30	GS1 DataBar Stacked Omnidirectional Composite			
31	HIBC Code 128 for LIC or PAS			+H123ABC0123456789 0D
32	HIBC Code 39 for LIC or PAS			+/EAH783B
33	HIBC CodaBlock-F for LIC or PAS			+/EAH783/Z34H159\$
34	HIBC Micro PDF417 for LIC or PAS			
35	IATA 2 of 5 Barcode	Yes		
36	Industrial 2 of 5 Barcode	Yes		
37	Interleaved 2 of 5 Barcode	Yes		
38	ISBN or International Standard Book Number		Yes	3161484100
39	ISMN or International Standard Music Number		Yes	M-2306-7118-7
40	ISSN or International Standard Serial Number		Yes	0264-3596
41	ITF-14 or UPC Shipping Container Symbol	Yes		
42	JAN 13		Yes	
43	JAN 8		Yes	
44	Logmars			
45	Matrix 2 of 5 Barcode	Yes		
46	MSI/Plessey			
47	Numly Number or ESN			1234567890123456789
48	Optical Product Code		Yes	123456789
49	Pharmacode One-Track			
50	Pharmacode Two-Track			
51	Pharma-Zentral-Nummer			123456
52	SCC-14 or Shipping Container Code			
53	SSCC-18 or UPC-128 Shipping Container Code			
54	Telepen Alpha			
55	Telepen Numeric			
56	UK Plessey			
57	UPC-A		Yes	1234567890
58	UPC-E		Yes	1234567
59	VICS BOL or VICS Bill of Lading			
60	VICS SCAC PRO			

Postal Code Barcodes				
61	Australia Postal Standard Customer			
62	Australia Postal Redirection			
63	Australia Postal Reply Paid			
64	Australia Postal Routing			
65	China Postal Code			
66	Danish Postal Code			CC12345678
67	Deutsche Post Identcode			12345678901
68	Deutsche Post Leitcode			1234567890123
69	France Postal Code 39			RA12345678
70	Italy Postal Code 2/5			
71	Italy Postal Code 39			
72	Japan Postal Code			1234567AZ
73	KIX or Netherlands Postal Code			A12345678
74	Korean Postal Code			123456
75	Royal Mail 4 State			
76	Singapore Postal Code			
77	Swiss Parcel Post Barcode			
78	USPS DAFT Code			DAFTTFAD
79	USPS Facing Identification Mark			A
80	USPS Horizontal Bars			
81	USPS OneCode or USPS Intelligent Mail			12345678901234567890 +50309
82	USPS PLANET			
83	USPS POSTNET			
84	USPS Sack Label			50309123
85	USPS Tray Label			5030912345
2D Barcodes				
86	Codablock-F			
87	Code 16K			
88	Code 49			
89	Code One			
90	Micro PDF417			
91	Micro QRCode			

8 Convert Size from CMs to Pixels

Internally our barcode Win32 DLL converts the value of barcode related size such as bar height and narrow bar width from centimeters to pixels based on device resolution, round up or round down float pixel value to the nearest integer.

The centimeter to pixel conversion formula is :

$$size_in_pixels = size_in_centimeters * device_resolution / 2.54$$

For example, if you render barcode on computer screen and the screen resolution is 96dpi.

(1) If bar height is 1.5 CMs, $\text{size_in_pixels} = 1.5 * 96 / 2.54 = 56.6929$, round up 56.6929 to 57, so actual bar height is 57 pixels.

(2) If bar height is 1.52 CMs, $\text{size_in_pixels} = 1.52 * 96 / 2.54 = 57.4488$, round down 57.4488 to 57, so actual bar height is 57 pixels.

(3) If bar height is 1.54 CMs, $\text{size_in_pixels} = 1.54 * 96 / 2.54 = 58.2047$, round down 58.2047 to 58, so actual bar height is 58 pixels.

Different bar height values might end up with same bar height in pixels due to performing rounding operations, same rounding operation is applied to narrow bar width conversion.

9 License

License agreement

This License Agreement ("LA") is the legal agreement between you and MW6 Technologies, Inc. ("MW6") for the font, and any electronic documentation ("Package"). By using, copying or installing the Package, you agree to be bound by the terms of this LA. If you don't agree to the terms in this LA, immediately remove unused Package.

1. License

* The Single User License allows the use of the software on **ONE** computer by **ONE** person in your organization.

* The Single Developer License allows 1 developer in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties, **each individual developer requires a separate Single Developer License as long as he or she needs access to MW6's product(s) and document(s).**

* The 2 Developer License allows 2 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The 3 Developer License allows 3 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The 4 Developer License allows 4 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The 5 Developer License allows 5 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The Unlimited Developer License allows unlimited number of developers in your organization the royalty-free distribution (unlimited number of users) of the software to the third parties.

2. User Disclaimer

The software is provided "as is" without warrant of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or noninfringement. MW6 assumes no liability for damages, direct or consequential, which may result from

the use of the software. Further, MW6 assumes no liability for losses caused by misuse or abuse of the software. This responsibility rests solely with the end user.

3. Copyright

The software and any electronic documentation are the proprietary products of MW6 and are protected by copyright and other intellectual property laws.
