

Tutorial for Crystal Reports Barcode Font Encoder UFL (U25IDAutomation.dll) - Deprecated

Crystal Reports Barcode Font Encoder UFL Overview

The IDAutomation Crystal Reports Linear Barcode Font Encoder UFL is very easy-to-use when generating barcodes in Crystal Reports. This UFL encoder tool supports many linear barcode types including Code 128, GS1-128, Code 39, Interleaved 2 of 5, UPC, EAN, Postnet, Intelligent Mail and more.

The GS1 DataBar functions are included in their own independent UFL, which are deprecated and no longer supported.

Linear UFL

DataBar UFL

User Function Library Guide		
UFL Type	Barcode Type (Symbology)	Supported Version of Windows**
Linear UFL	Linear: Code 11 , Code 128 , Code 39 , Code 93 , Codabar , EAN-13 , EAN-8 , Interleaved 2 of 5 , MSI Plessey , UPC-A , UPC-E , USPS IntelligentMail , USPS Postnet , and USPS Planet	x86 and x64
	Linear DataBar (Deprecated): DataBar , DataBar Expanded , DataBar Expanded Stacked , DataBar Limited , DataBar Stacked , DataBar Stacked Omnidirectional , and DataBar Truncated	x86
DataBar UFL	DataBar: DataBar , DataBar Expanded , DataBar Expanded Stacked , DataBar Limited , DataBar Stacked , DataBar Stacked Omnidirectional , and DataBar Truncated	x86 and x64

** 32 bit = 'x86', 64 bit = 'x64'. When using a Crystal Reports Redistributable version, IDAutomation does not recommend 'AnyCPU' compilations. Per [SAP](#), Crystal Reports Designer runs only in 32 bit mode.

Crystal Reports Linear Barcode UFL Installation

Crystal Reports Designer is a 32-bit program and runs only in 32-bit mode, even on 64-bit systems. The UFL setup will install the respective UFL to the following Windows System** directory:

- 32-bit Operating System: %windir%\System32\
- 64-bit Operating System: %windir%\SysWOW64\

If the IDAutomation functions do not show up in the *Additional Functions* section after installing the UFL, copy the U25IDAutomation.dll file from the Windows System directory (for example C:\Windows\System32) to the directory where the Crystal DLLs are located.

- **In Crystal 11** the file may be copied to C:\Program Files\Common Files\Business Objects\3.0\bin;
- **In Crystal 10** and previous versions the file may be copied to %Windows%\Crystal where %Windows% is the directory where Windows is installed.

Crystal Reports for Visual Studio Runtime Distribution: Please see the [UFL & Font Distribution with Crystal Runtime Files](#) section of the manual. In the UFL download, there are folders labeled 32-bit and 64-bit to distinguish the DLL runtime of each file.

*** %windir% is the Windows directory, which can be found by pressing Windows + R > type in %windir% > Enter*

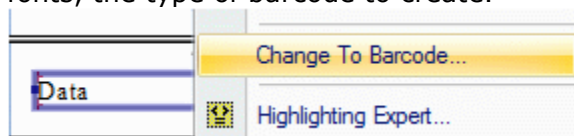
Notes:

Although [DataBar functions are present in the Linear UFL](#) for backwards compatibility, they are only compatible with 32 bit versions of Windows. For compatibility with both 32-bit and 64-bit, IDAutomation recommends their new [DataBar UFL](#).

'Change to Barcode' Functionality for Crystal Reports 2008 & Above

Turning fields into barcodes is a new functionality introduced in Crystal Reports 2008 (Version 12). When using Crystal versions below 12, the [standard UFL tutorial](#) must be followed. This feature is exclusively available for the Linear UFL.

1. Install the appropriate [barcode fonts](#). If in doubt about which font to use, IDAutomation recommends the [Universal Barcode Font](#) for the Code 128 barcode.
2. Download and install the latest version of [IDAutomation's Crystal Reports Linear UFL](#). Only versions released March 2008 or later are compatible with the *Change to Barcode* functionality introduced in Crystal Reports 12.
3. Add a field to the report that contains the data that will be encoded in the barcode. If necessary, a [formula field](#) may be created to append text or concatenate fields.
4. Select the *Preview* tab (or *File - Print Preview*) to confirm that the correct data appears in the field.
5. Right-click on this field, choose *Change to Barcode* and select from the list of installed barcode fonts, the type of barcode to create.



6. The barcode should be instantly created in the report. To create a text interpretation, just append the same field to the report again and position it below the barcode as necessary.
7. Print one page of the report and test with a [barcode scanner](#).

Linear Barcode UFL & Font Distribution with Crystal Runtime Files

IDAAutomation's Linear UFL is free to use and distribute in an application with the purchase of any IDAutomation fonts package. The following runtime files should be distributed with the application. Although DataBar functions will be present in this UFL, it is not recommended to be used on 64-bit systems. Instead, refer to the dedicated [DataBar UFL](#).

Linear UFL Files to Distribute with Crystal Runtimes: 32-bit Operating System		
Type	File to Distribute	Notes
Font File(s)	???.ttf	TrueType fonts may be distributed with the purchase of a Developer License.
32-bit Crystal UFL	\32bit\U25IDAAutomation.dll	The IDAutomation 32-bit Crystal UFL should be installed in the Windows System* directory (%windir%\System32)

Linear UFL Files to Distribute with Crystal Runtimes: 64-bit Operating System			
Type	File to Distribute	Notes: Although present, do not use DataBar functions - use the DataBar UFL instead.	Compiled Solution Platform
Font File(s)	???.ttf	TrueType fonts may be distributed with the purchase of a Developer License.	x86 and x64
32-bit Crystal UFL	\32bit\U25IDAAutomation.dll	The IDAutomation 32-bit Crystal UFL should be installed in the 'Windows 32-bit on Windows 64-bit' System directory (%windir%\SysWOW64)	x86
64-bit Crystal UFL	\64bit\U25IDAAutomation.dll	The IDAutomation 64-bit Crystal UFL should be installed in the Windows System directory (%windir%\System32)	x64

** %windir% is the Windows directory, which can be found by pressing Windows + R > type in %windir% > Enter

Crystal UFL Barcode Functions List

The functions listed below are all available in the Crystal Reports UFL download are valid when used with the barcode font listed in the *Font to Use* column. The data input type for all functions is a string.

Functions for Standard Barcode Fonts		
Barcode Type	Functions (View Function Property Descriptions)	Font to use
Code 11	IDAutomation Code 11(DataToEncode)	IDAutomationC11
Code 128	IDAutomation Code128(DataToEncode) <i>Recommended for general barcoding purposes.</i>	IDAutomationC128
	Additional Code 128 Functions IDAutomation Code128ApplyTilde (DataToEncode) <i>IDAutomation_Code128ApplyTilde() applies Code 128 tilde processing to the barcode and is used to create GS1-128 or encode functions.</i>	IDAutomationC128
	IDAutomation Code128A(DataToEncode) IDAutomation Code128B(DataToEncode) IDAutomation Code128C(DataToEncode)	
	IDAutomation_Code128HumanReadableTilde (DataToEncode) <i>This method returns the human readable text for Code 128 barcodes, such as with GS1-128.</i>	Text Font
Code 39	IDAutomation Code39(DataToEncode) IDAutomation Code39Mod43(DataToEncode)	IDAutomationC39
Code 93	IDAutomation Code93(DataToEncode)	IDAutomationC93
Codabar	IDAutomation Codabar(DataToEncode)	IDAutomationCB
EAN-13	IDAutomation EAN13(DataToEncode)	IDAutomationUPCEAN
EAN-8	IDAutomation EAN8(DataToEncode)	IDAutomationUPCEAN
Interleaved 2 of 5	IDAutomation I2of5(DataToEncode) IDAutomation I2of5Mod10(DataToEncode)	IDAutomationI25
MSI Plessey	IDAutomation MSI(DataToEncode)	IDAutomationMSI

UPC-A	IDAAutomation_UPCa(DataToEncode)	IDAAutomationUPCEAN
UPC-E	IDAAutomation_UPCe(DataToEncode)	IDAAutomationUPCEAN
USPS IntelligentMail	IDAAutomation_IntelligentMail(DataToEncode)	IDAAutomationPOSTNET or IDAAutomationIMB
USPS Postnet	IDAAutomation_Postnet(DataToEncode)	IDAAutomationPOSTNET
USPS Planet	IDAAutomation_Postnet(DataToEncode)	IDAAutomationPOSTNET
<i>Not Applicable</i>	IDAAutomation_Mod10(DataToEncode)	<i>not applicable</i>

Universal Barcode Font Functions

The functions listed below are preceded with [IDAAutomation_Uni_](#) and should be used with [IDAAutomation Universal Barcode Fonts](#) or [IDAAutomation DataBar Barcode Fonts](#).

Barcode Type	Functions	Font to Use
Code 128	IDAAutomation_Uni_C128(DataToEncode) <i>C128() is recommended for general barcoding purposes.</i>	IDAAutomation_Uni or IDAAutomation DataBar
	Additional Code 128 Functions IDAAutomation_Uni_C128ApplyTilde(DataToEncode) <i>IDAAutomation_Uni_C128ApplyTilde() applies Code 128 tilde processing and is used to create GS1-128 or encode functions.</i>	IDAAutomation_Uni or IDAAutomation DataBar
	IDAAutomation_Uni_C128A(DataToEncode) IDAAutomation_Uni_C128B(DataToEncode) IDAAutomation_Uni_C128C(DataToEncode)	
	IDAAutomation_Code128HumanReadableTilde (DataToEncode) <i>This method returns the human readable text for Code 128 barcodes, such as with GS1-128.</i>	Text Font
Code 39	IDAAutomation_Uni_C39(DataToEncode, N Dimension, IncludeCheckDigit) Example: <code>IDAAutomation_Uni_C39("12345678",3,TRUE)</code>	IDAAutomation_Uni or IDAAutomation DataBar
Codabar	IDAAutomation_Uni_Codabar(DataToEncode, N Dimension, StartChar, StopChar) Example:	IDAAutomation_Uni or IDAAutomation DataBar

	<code>IDAAutomation_Uni_Codabar("12345678",2,A,B)</code>	
Interleaved 2 of 5	IDAAutomation_Uni_I2of5(DataToEncode, N_Dimension, IncludeCheckDigit) Example: <code>IDAAutomation_Uni_I2of5("12345678",2,FALSE)</code>	IDAAutomation_Uni or IDAutomation DataBar
MSI Plessey	IDAutomation_Uni_MSI(DataToEncode, N_Dimension, IncludeCheckDigit)	IDAutomation_Uni or IDAutomation DataBar
USPS IntelligentMail	IDAutomation_Uni_IntelligentMail(DataToEncode)	IDAutomation_Uni
USPS Postnet	IDAutomation_Uni_Postnet(DataToEncode, IncludeCheckDigit)	IDAutomation_Uni
USPS Planet	IDAutomation_Uni_Planet(DataToEncode, IncludeCheckDigit)	IDAutomation_Uni

Linear DataBar Functions (Deprecated- 32 bit Windows Only)

Although the Linear DataBar functions are deprecated (present in the Linear UFL for backwards compatibility), they are only compatible with 32-bit versions of Windows. **IDAutomation recommends the use of the DataBar UFL for any implementation being run on either 32-bit or 64-bit versions of Windows.**

DataBar Functions		
The methods listed below are preceded with <code>IDAutomation_DataBar_</code> and are only to be used with the IDAutomation GS1 DataBar Font .		
Barcode Type	Functions	Font to use
DataBar	IDAutomation_DataBar(DataToEncode)	IDAutomation DataBar 34
DataBar Expanded	IDAutomation_DataBarExpanded(DataToEncode)	IDAutomation DataBar 34
DataBar Expanded Stacked	DataBar Expanded Stacked requires a custom formula field to be copied from the sample report provided.	IDAutomation DataBar 34

DataBar Limited	IDAutomation DataBarLimited(DataToEncode)	IDAutomation DataBar 13
DataBar Stacked	IDAutomation DataBarStacked(DataToEncode)	IDAutomation DataBar 13
DataBar Stacked Omnidirectional	IDAutomation DataBarStackedOmnidirectional(DataToEncode)	IDAutomation DataBar 34
DataBar Truncated	IDAutomation DataBar(DataToEncode)	IDAutomation DataBar 13

Linear DataBar Functions (Deprecated- 32 bit Windows Only)

Although the Linear DataBar functions are deprecated (present in the Linear UFL for backwards compatibility), they are only compatible with 32-bit versions of Windows. **IDAutomation recommends the use of the [DataBar UFL](#) for any implementation being run on either 32-bit or 64-bit versions of Windows.**

DataBar Functions		
The methods listed below are preceded with <code>IDAutomation_DataBar_</code> and are only to be used with the IDAutomation GS1 DataBar Font .		
Barcode Type	Functions	Font to use
DataBar	IDAutomation DataBar(DataToEncode)	IDAutomation DataBar 34
DataBar Expanded	IDAutomation DataBarExpanded(DataToEncode)	IDAutomation DataBar 34
DataBar Expanded Stacked	DataBar Expanded Stacked requires a custom formula field to be copied from the sample report provided.	IDAutomation DataBar 34
DataBar Limited	IDAutomation DataBarLimited(DataToEncode)	IDAutomation DataBar 13

DataBar Stacked	IDAutomation DataBarStacked(DataToEncode)	IDAutomation DataBar 13
DataBar Stacked Omnidirectional	IDAutomation DataBarStackedOmnidirectional(DataToEncode)	IDAutomation DataBar 34
DataBar Truncated	IDAutomation DataBar(DataToEncode)	IDAutomation DataBar 13

Function Property Descriptions

- **DataToEncode:** a string value that represents the data being encoded.
- **N_Dimension:** determines the width of the wide bars that is a multiple of the X dimension. Valid values are 2, 2.5 and 3. The default is 2. The X dimension is determined by the font point size.
- **IncludeCheckDigit:** a Boolean value that determines whether a check digit should be automatically calculated and included for the DataToEncode.
- **ApplyTilde:** when ApplyTilde is set to "1" in Code 128 with the Auto character set, the format ~ddd may be used to specify the ASCII code of the character to be encoded and [several other tilde options are enabled](#).

Font Formulas

Many of Automation's font packages include [Font Formulas](#), which allow font formatting routines to be embedded in a report without the need for a UFL. These formulas are saved as part of the report file (.rpt) and do not have any external dependencies, with the exception of the required barcode font. When distributing reports, consider this method or try the [Native Barcode Generator for Crystal Reports](#), which creates dynamic embedded barcodes without fonts.