

CompuArt

C.P. 5541

16158 - GENOVA Voltri - ITALY

Fax +39 10 6530179 - +39 10 6513199

www.compuart.it

Summary

[Introduction](#)

[Rapid show to the program](#)

[Use of the program](#)

[The bar codes](#)

[Copy and Paste operations](#)

[Automatic requests to CompuArt by e-mail](#)

[Link to other programs](#)

[Creation of labels using Label Laboratory](#)

[Registration](#)

[Differences between the shareware and the professional version](#)

[Associations for the distribution on EAN prefixes](#)

Introduction

This program gives you the possibility to insert, in a very simple way, bar codes in every type of document. The bar codes at your disposition are very common so all bar code scanners can recognize them. You can choose among a large number of options: numeric or alphanumeric codes, fixed or variable length, with or without check character. Its possible, for every type of code, to set the essential parameters (bars dimensions, print settings, font, etc.) to personalize it as you like.

The function "**Print the bar code**" from the setting window, allows you to control immediately the results of your modifications.

You dont have any difficulty to define the characteristics, even if you dont know anything about the bar codes; in fact, an automatic system will follow you showing you texts and explication pictures for every value that you want to modify. Pressing **F1**, a guide gives you information about the code you are creating.

This program contains a particular function that allows you to reduce the bar thickness: in this way you will obtain clear prints using every type of printer, also for code printed with a very high density. In fact, this method reduces the slaver of the black over the established limits: this lack is present in all the printers (i.e., the bars dimension, besides the black intensity, changes using a new tape or an old one). The aim of the program is to create bar codes, copy them in the Windows clipboard and from here they can be paste in destination application. To create the code, the program uses Windows graphic metafiles. This particular form is specific of the operative system because it collects the standard calles of the Windows graphic interface: its because the program is compatible with all the applications and its very precise (hundredth of millimeter).

Its also possible to save the bar code on file: the generated file is a graphic metafile with an heading (Aldus placeable).

You can use CreaBar with other applications to create labels, to mark the dealing products, to manage the stores, to plan the packing: these are the classic applications of the bar code. However, we have planned Creabar with many characteristics to give an incentive to the documents automation. We should like to help you to realize systems that help to speed up the manipulation of large quantities of paper. You can insert the bar code in all the documents you give to other persons and return to be registered (i.e., current account bulletins, advertisements, coupons, payment receipts of your clients, registration of the payments about issued invoices).

Using the dynamic link libraries GBCODE.DLL and GBCODE32.DLL, you can use CreaBar with other programs that have a programmation or a macro language (as Excel) and with specific programs as our Label Laboratory (for the creation of labels) and WinCCP (for the print of modules).

Our other programs will be modified to accept bar codes: we are sure you give us useful advices to develop and improve our work and we thank you in advance.

[Rapid show to the program](#)

[Use of the program](#)

[Summary](#)

Use of the program

The peculiarity of CreaBar, is the possibility to set the characteristics of the bar code that you want to create. In fact, its possible to establish the height and the thickness of the bar, the type of the text font written under the code (if it must to be printed), the eventual magnification factor, etc.

The bar codes parameters are memorized in files with .BRC extension. In practice, you can create different files with different settings using the same type of bar code (EAN 8, EAN 13, etc.). For example: you want to create two different types of EAN13 code, one with magnification 90% and the other with magnification 150%. You can create these files using the menu option "**New bar code configuration**" and give them different names, i.e., "EAN13-90.BRC" and "EAN13150.BRC".

The new look of CreaBar is similar to Windows 95 characteristics.

The main functions are available from some speed buttons and from the menu.

The main functions are:

- Load a bar code configuration
- Create a new bar code configuration
- Modify the bar code configuration
- Save the bar code on file
- Print the bar code
- Exit

Before to create a bar code, its necessary to load a relative configuration file. When the program is installed, it creates automatically a file called EAN8.BRC.

To load the files of the bar code parameters, you can use the "speed button" or the option "Load a bar code configuration" from the menu "File".

You must write the text to codify in the input line.

The number of characters to insert depends on the type of used code. The program suggests how many characters to insert if you use fixed length codes, as EAN and UPC.

When its necessary, the check character is calculated and added automatically by the program.

When the characters to codify are inserted, press by the mouse the button "Copy" to the right of the input line or press the key "**Return**" to copy the bar code in the Windows clipboard.

The created code will appear in the space under the input line.

An error message will advise you if the number of characters is wrong.

Its possible to set the measurement unit of the bar codes in millimeters or in inches directly from the window "Creation/modification of the bar code" selecting your preference from the menu "Options".

Now the code can be imported in the application that you want, using the function "**Paste**" from the menu or the key **SHIFT+INS** or **CTRL+V** according to the program.

Summary

The bar codes generated by the program

The bar codes generated by CreaBar 2.0 are:

EAN 8 - Eight numeric characters, fixed length with check character.

EAN 13 - Thirteen numeric characters, fixed length with check character.

UPC/A - Twelve numeric characters, fixed length with check character.

UPC/E - Six numeric characters, fixed length with check character.

EAN/UCC 128 - A variation of Code 128 used to codify wares, variable length with automatic check character.

CODE 39 - 43 alphanumeric characters with seven special characters, variable length with optional check character.

CODE 39 FULL ASCII - Extension of Code 39: it allows to codify all the ASCII characters (128), variable length with optional check character.

CODE 93 - It allows you to codify all the ASCII characters, it produces a code more compact than Code 39, variable length with automatic check character.

CODE 128 - It allows to codify all the ASCII characters, variable length with automatic check character.

CODE 32 (Italian Pharmaceutical) - Nine numeric characters, fixed length with check character.

INTERLEAVE 2/5 - Numeric characters, variable length with optional check character.

CODABAR - Numeric characters plus six special characters, variable length with optional check character.

Summary

Code EAN/UCC 128

Its a variation Code 128, reserved exclusively to EAN International and UCC (Uniform Code Council). The aim of this code is to establish a standard to label the goods in a more flexible way, giving lots information about an only product code; the retail sales cannot use this code.

EAN/UCC 128 is one of the most complete alphanumeric one-dimension codes actually available. It can codify all the 128 ASCII characters in compact form, as Code 128.

Its different from Code 128 because it always contains the function character "FNC1" that follows the start character of the code. The other function characters "FNC2, FNC3, FNC4" are not available.

To obtain a "minimization" of the code lenght, the program inserts automatically the character "FNC1" and makes an analysis of the text to codify to use the special characters Start, Code A-B-C and Shift.

To insert the check characters in the text, they must be codified in a particular way, as for Code 39 FULL ASCII.

To insert not printable characters in the text to codify, you should write the symbol ## before the ASCII value. For example, if you want to codify the following text:

Compu<return>Art

the character <return> is not printable, so it cannot be inserted in the text. Consulting the ASCII table, you can see that this is a check character corresponding to the value 13, so the string to insert is the following one:

Compu##013Art

The character ## should be followed by three numbers so, if you have a number composed by one or two ciphers, you must insert one or two zero before it.

Code EAN/UCC 128 contains several data: the product identification code, the expiration date, the product dimensions, the series number.

To have more information about Code EAN/UCC 128, we advise you to consult the documentation on Internet, for example in this site: <http://www.uc-council.org>

The request information to personalize this code are the following ones:

Thickness of the narrow bar

Bar code height

Bar width reduction

Print text below the code

Choose the text font

Summary

The bar codes generated by the program

Code 128

Code 128 can codify all the 128 ASCII characters. It has variable length, automatic check character and it codifies numeric characters in a double density compact form.

Its a self-checking, bidirectional, continuous code. Every character is represented by 6 elements subdivided in 3 bars and 3 spaces.

To obtain a minimization of the text length, the program executes an analysis of the text to codify to use the special characters Start, Code A-B-C and Shift.

To insert the check characters in the text, they must be codified in a particular way, as for Code 39 FULL ASCII.

To insert not printable characters in the text to codify, you should write the symbol ## before the ASCII value. For example, if you want to codify the following text:

Compu<return>Art

the character <return> is not printable, so it cannot be inserted in the text. Consulting the ASCII table, you can see that this is a check character corresponding to the value 13, so the string to insert is the following one:

Compu##013Art

The character ## should be followed by three numbers so, if you have a number composed by one or two ciphers, you must insert one or two zero before it.

You can codify the function characters "FNC1, FNC2, FNC3, FNC4" inserting in the text the values "##201, ##202, ##203, ##204".

To personalize this code the request information are the following ones:

Thickness of the narrow bar

Bar code height

Bar width reduction

Print text below the code

Choose the text font

Summary

The bar codes generated by the program

Thickness of the narrow bar

The minimum width of Code 128 narrow bars is 0.19 mm. This value is used to obtain high density prints. If you have a laser print, set the width of the bars to 0.19 mm and, in the field "Bar width reduction", insert a value to obtain clear bars, clearly outdistanced the ones from the others: so you obtain a code printed in high resolution.

If you use dot matrix printers this value must be higher than 0.19 to obtain an acceptable result.

Code 128

Code EAN/UCC 128

Bar code height

For Code 128, the standard value of the bars minimum height should be at least 6.5 mm.

Remember that if the code is too long the lecture is very difficult: if you use bar code wands, set the height code at least to 15% of its length.

When you use laser scanners the code should be greater: the advisable value is 20 mm or 20% of the code length.

Code 128

Code EAN/UCC 128

Code 39

Code 39 is one of the most common bar codes. It can codify until 43 symbols: alphabetical capital characters from A to Z, numeric characters from zero to nine and the following special characters: "_" "." " " "\$" "/" "+" "%".

Its a self-checking, bidirectional code that has variable length.

Every character is represented by 9 elements subdivided in 5 bars and 4 spaces.

This code is not much compact for its particular codification system, so its not qualified for very high information density print.

Its regulated by the norms ANSI MH10,8 M-1983 and MIL-STD-1189.

To personalize this code the request information are the following ones:

Thickness of the narrow bar

Narrow to wide bar ratio

Bar code height

Thickness of intercharacter space

Bar width reduction

Include check character

Print text below the code

Choose the text font

Summary

The bar codes generated by the program

Code 39 FULL ASCII

This code is an extension of Code 39 and maintains the same characteristics: it can codify all the 127 ASCII characters.

To insert not printable characters in the text to codify, you should write the symbol **##** before the ASCII value. For example, if you want to codify the following text:

CPP<return>GH

the character <return> is not printable, so it cannot be inserted in the text. If you consult the ASCII table, you can notice that this is a check character corresponding to the value 13. The string to insert is the following one:

CCP##013GH

The character **##** should be followed by three numbers so, if you have a number composed by one or two ciphers, you must insert one or two zero before it.

To personalize this code the request information are the following ones:

Thickness of the narrow bar

Narrow to wide bar ratio

Bar code height

Thickness of intercharacter space

Bar width reduction

Include check character

Print text below the code

Choose the text font

Summary

The bar codes generated by the program

Thickness of the narrow bar

The minimum width of Code 39 narrow bars is 0.19 mm. This value is used to obtain high density prints. If you have a laser print, set the width of the bars to 0.19 mm and, in the field "Bar width reduction", insert a value to obtain clear bars, clearly outdistanced the ones from the others: so you obtain a code printed in high resolution.

If you use dot matrix printers this value must be higher than 0.19 to obtain an acceptable result.

Code 39

Code 39 FULL ASCII

Narrow to wide bar ratio

Code 39 contains only two types of bars: narrow and wide. The ratio between these two types of bars can change from 2:1 to 3:1.

[Code 39](#)

[Code 39 FULL ASCII](#)

Bar code height

For Code 39, the standard value of the bars minimum height should be at least 6.5 mm.

Remember that if the code is too long the lecture is very difficult: if you use bar code wands, set the height code at least to 15% of its length.

When you use laser scanners, the code should be greater: the advisable value is 20 mm or 20% of the code length.

Code 39

Code 39 FULL ASCII

Thickness of intercharacter space

In Code 39 every codified character is separated from another by a space. The value of this space can be set from one to three times the narrow bar thickness.

Code 39

Code 39 FULL ASCII

Include check character

According to the used reader, you can add to the codified string, a check character to verify the precision of the text during the lecture.

Code 39

Code 39 FULL ASCII

Code 93

This code has been created by Intermec after Code 39 and it can contain an higher information density than the previous one.

It can codify all the 127 ASCII characters. Its very precise because it has two check characters added automatically.

To insert the check characters in the text, they must be codified in a particular way, as for Code 39 FULL ASCII.

To insert not printable characters in the text to codify, you should write the symbol ## before the ASCII value. For example, if you want to codify the following text:

Compu<return>Art

the character <return> is not printable, so it cannot be inserted in the text. Consulting the ASCII table, you can see that this is a check character corresponding to the value 13, so the string to insert is the following one:

Compu##013Art

The character ## should be followed by three numbers so, if you have a number composed by one or two ciphers, you must insert one or two zero before it.

To personalize this code the request information are the following ones:

Thickness of the narrow bar

Bar code height

Bar width reduction

Print text below the code

Choose the text font

Summary

The bar codes generated by the program

Thickness of the narrow bar

The minimum width of the Code 93 narrow bars is 0.2 mm. This value is used to obtain high density prints. If you have a laser print, set the width of the bars to 0.2 mm and, in the field "Bar width reduction", insert a value to obtain clear bars, clearly outdistanced the ones from the others: so you obtain a code printed in high resolution.

If you use dot matrix printers this value must be higher than 0.2 to obtain an acceptable result.

Code 93

Bar code height

For Code 93, the standard value of the bars minimum height must be at least of 6.5 mm.

Remember that if the code is too long the lecture is very difficult: if you use bar code wands, set the height code at least the 15% of its length.

When you use laser scanners, the code should be greater: the advisable value is 20 mm or 20% of the code length.

Code 93

Bar width reduction

According to the used printer, the print of the black bar can slaver over the established limits. In the high resolution print, it can happen that the bars are too near the ones to the others, so the black spaces and the white ones dont respect the standard value: in this case the code is not readable. The parameter "Bar width reduction" is used to limit this problem. This number is subtracted from the left and from the right side of the black bar so it becomes thinner that before. The used value must be tried according to the used printer.

An indicative value for dot matrix printers can be 0.1 mm.

Bar codes

Print text below the code

According to the used code, it can be useful to print the codified text string with the bar code. Its possible to choose the font among all the types available from Windows.

The string must not exceed the right side of the code, so pay attention to the dimension of the used font because the exceeded part of string is not printed.

Bar codes

Measurement units

Its possible to select, from the menu "Options", the unit of measure used to show the values in the window of bar code creation/modification. The available units are millimeters or inches. The conversion of the inserted values is executed automatically.

Use of the program

EAN8 EAN13 UPC/A UPC/E

Code EAN is the most common code. We can find it in the package of all the products we buy. Its composed by 8 or 13 ciphers; one of them is the check character. These ciphers must respect particular rules because this code is an international standard that has the aim to give a univocal number to every product.

Reading code **EAN13** from left to right, we have:

- **2** ciphers to codify the product origin nation;
- **5** ciphers to codify the producer. Every producer has a number assigned, for Italy, by **INDICOD** (Milan).
[Click here for other Nations product number](#)
- **5** ciphers to codify the product. Every producer assigns a univocal number to his product;
- **1** cipher is the check character used to verify the validity of the lecture.

It's also possible to codify loose products sold to weight or packages with variable weight, i.e., the labels of the balances in the supermarkets. In this case the codify is:

- **2** ciphers to specify that the product is loose. The available numbers are from 20 to 29;
- **5** ciphers to codify the product code or a progressive number;
- **5** ciphers to codify the price;
- **1** cipher is the check character.

Code **UPC/A** (Universal Product Code) is equivalent to code **EAN13** and it is used in USA, Canada and United Kingdom. The ciphers, excluded the first two ones, have the same meaning of code **EAN13**. The first cipher is zero to indicate the codify UPC, the second one is the System Number that has the following meaning:

- **0** normal use of the code;
- **2** loose products;
- **3** used for the National Drug Code and National Health Related Items Code;
- **4** used for no alimentary products;
- **5** used in the coupons.

Code **EAN8** is used in the packages that cannot use code **EAN13** for limit of space.

- **2** ciphers to codify the nation;
- **4** ciphers to codify the producer;
- **1** cipher to codify the product;
- **1** cipher is the check character.

Code **UPC/E**, as code **EAN8**, is used to sign products that have limits of dimensions and cannot use **UPC/A**. This is a zero suppressed codify because its created subtracting many zero from a code **UPC/A**. In all these codes, a supplementary codify or added code is available: it can be composed by two or five characters.

To personalize this code the request information are the following ones:

[Thickness of the narrow bar](#)

[Bar code height](#)

[Magnification factor](#)

[Bar width reduction](#)

[Include check character](#)

[Print text below the code](#)

[Choose the text font](#)

[Summary](#)

[The bar codes generated by the program](#)

Thickness of the narrow bar

The standard thickness of EAN narrow bar is 0.33 mm. If this value is unchanged, its possible to enlarge or restrict the code using the "Magnification factor".

This value can be used to create personalized codes. The value of the narrow bar that will be printed, will be calculated using the following ratio:

Thickness of the narrow bar = (set value) x Magnification factor / 100

If the black bars are not clear respect the white ones, you should increase the value "Bar width reduction".

Codes EAN

Bar code height

This value specifies the height of the bars.

The start, the stop and the central bars are longer than the other ones, for a value that is five times the thickness of the narrow bar.

The height of these bars is automatically calculated by the program and the value shown in the "Bar code height" window is relative to the shorter bars.

The definitive height of the bars is calculated with the following ratio:

Height bars = (set value) x Magnification factor / 100

Codes EAN

Magnification factor

The default values of code EAN can be increased or restricted, multiplying them by a percentage value from 80% to 200%.

Codes EAN

Include check character

The last character of codes EAN and UPC is a check character that must always be inserted. Selecting this option, the program calculates automatically this character according to the code data.

If you want to insert the entire code, included the check character, you must deselect this option: in this case the program doesn't execute the control of the code precision.

Codes EAN

Print text below the code

In the standard EAN, you must print the text below the bar code. You can deselect the automatic text print, if you need a very compact personalized code or you want to create labels with the text positioned in different ways.

Codes EAN

Choose the text font

Its possible to choose the font of the text to print below the bar code.

Its necessary to choose a character dimension that doesnt exceed the lenght of the bar code.

In code EAN, the height of the characters is modified during the print according to the value of the "Magnification factor".

Bar codes

Code 32 (Italian Pharmaceutical)

Its an extension of Code 39 and its used to codify the medicines package in Italy. The code is preceded by the letter A that is not codified in the bars: it indicates the speciality of the medicine.

This code is composed by eight ciphers plus the check character. The nine numeric ciphers are codified in bars using six alphanumeric ciphers of Code 39, thanks to a particular transcodification formula. In this way the code is more compact.

To personalize this code the request information are the following ones:

Thickness of the narrow bar

Narrow to wide bar ratio

Bar code height

Bar width reduction

Print text below the code

Choose the text font

Summary

The bar codes generated by the program

Thickness of the narrow bar

The standard thickness of Code 32 narrow bar is between 0.25 and 0.254 mm.

Code 32

Narrow to wide bar ratio

Code 32 contains only two types of bars: narrow and wide. The standard ratio between these two types of bars is 2.5:1 but it can change from 2:1 to 3:1.

Code 32

Bar code height

For Code 32, the standard value of the minimum height of the bars must be at least 7.5 mm. Its possible to personalize the code giving this parameter different values.

Code 32

Print text below the code

You can decide if you want to print or not the text below the bar code.

The letter A comes before the number to codify that is followed by a check character.

Its possible to choose the font among all the types available from Windows.

The string must not exceed the right side of the code, so pay attention to the dimension of the used font because the exceeded part of string is not printed.

Code 32

Interleave 2/5 (ITF)

This code is very common and reliable. Its lecture is bidirectional, it has a variable length and it codifies only numeric characters. Its very compact and it codifies a very high density of information. The check character is optional.

Its possible that the line of the reader gets out from the code length before to encounter the start and the stop bars: it can happen for the particular conformation of this code and it causes a wrong lecture of the code. You can ward off this problem setting the decodificators to accept only a code fixed length or using the check character.

Many readers dont read strings that contain an uneven number of characters: in this case, during the fase of codification, the program adds automatically a zero before the first character to obtain an even length string.

To personalize this code the request information are the following ones:

Thickness of the narrow bar

Narrow to wide bar ratio

Bar code height

Bar width reduction

Include check character

Print text below the code

Choose the text font

Summary

The bar codes generated by the program

Thickness of the narrow bar

This code is codified in four elements: bars and narrow spaces, bars and wide spaces. The dimension of the narrow bar is defined, while the dimension of the wide bar is set only as a ratio. The total length of the code depends on the wide bar value. Remember to do many tests because it can happen that the printer cannot print very little codes.

Interleave 2/5

Narrow to wide bar ratio

When the print density is too high or the support can be deformed, the ratio between the narrow and the wide bar dimension, determines the readability of the code. It's better to increase this value to guarantee the reliability of the code. You should set values from 2.25:1 to 3:1 for narrow bars that have dimensions lower than 0.5 mm and values from 2:1 to 3:1 for dimensions higher than 0.5 mm.

Interleave 2/5

Bar code height

Its possible to set the bars height as you like if the code will be read by a pen type reader or a bar code laser scanner.

Interleave 2/5

Include check character

Its better, in order to increase the reliability of the data, to add to the codified string a check character. To obtain a right test the decodificator must be set to know this character as a check character.

Interleave 2/5

CODABAR

This code can codify the numbers from zero to nine and the following special characters: - \$: / . +. It has four special start/stop characters that are A B C D and they can be used to discern one code from another one. Before the text to codify in the input line you should write one of these special start characters, i.e., the number **345.56** must be inserted in this way:

A345.56 or **B345.56** or **C345.56** or **D345.56**.

The start character will be not written in the text below the code.

CODABAR is also a self-checking code, i.e., every character has a control bit but its also possible to insert a check character for the entire string.

Thickness of the narrow bar

Narrow to wide bar ratio

Bar code height

Thickness of intercharacter space

Bar width reduction

Include check character

Print text below the code

Choose the text font

Summary

The bar codes generated by the program

Thickness of the narrow bar

The minimum width of CODABAR narrow bars is 0.165 mm. This value is used to obtain high density prints. If you have a laser print, set the width of the bars to 0.165 mm and, in the field "Bar width reduction", insert a value to obtain clear bars, clearly outdistanced the ones from the others: so you obtain a code printed in high resolution.

If you use dot matrix printers this value must be higher than 0.165 to obtain an acceptable result.

CODABAR

Narrow to wide bar ratio

CODABAR contains only two types of bars: narrow and wide. The ratio between these two types of bars can change from 2:1 to 3:1. It's advisable, during high density prints, to increase, in the highest degree the difference between the two types of bars setting the value near three.

CODABAR

Bar code height

The minimum height of the code using pen type readers is 6.5 mm. For laser scanners the dimensions must be increased according to the distance of lecture.

CODABAR

Thickness of intercharacter space

In CODABAR every codified character is separated from another by a space. The value of this space can be set from one to five times the thickness of the narrow bar.

CODABAR

Include check character

CODABAR is a self-checking code, i.e., all the codified characters have a control bit that allows you to find eventual mistakes during the lecture. It's possible to add a check character after the last character of the code, to make the lecture more certain. The reader must be set to know this character.

CODABAR

Copy and Paste operations

Pressing the button "**Copy**", after the insertion in the input line of the text to codify, the generated bar code is copied in the **clipboard**.

The **clipboard** is a temporary file for texts or graphics cut and copied. Its possible to copy the content of the clipboard in every document or in other applications (i.e., Microsoft Word, Excel, Publisher, etc.).

The clipboard maintains the same information until you make another copy operation.

The clipboard content can be inserted in the current application or document using the command "Paste".

This command can be followed by a term that identifies the content of the notes (i.e., Paste object).

The command **Paste** is inactive if the clipboard is empty.

If the bar code is pasted in a Word Processor, it will be inserted where the pointer is positioned.

If it is pasted in a Desktop Publisher, it will be used like an object and inserted in the center of the window.

It's possible to put the object in every point of the window.

If it is pasted in an electronic table (i.e., Excel), it will be positioned starting from the selected cell.

Pay attention to the border, because the white space around the code must be sufficiently large to allow the reader to see the first bar and the other lines don't disturb it.

To paste the bar code in the document in a rapid way, you press contemporaneously CTRL + V or SHIFT + INS, when its allowed by the program.

Summary

Automatic requests to CompuArt by e-mail

The Internet users can ask directly to CompuArt the most frequent questions, if the version that they are using is the last one and eventually have the last version directly by e-mail.

They can be inserted in the Mailing List, in order to be informed about upgrades or new programs produced by CompuArt.

These requests will be developed by a dedicate computer in order to make the service faster.

Summary

Link to other programs

Attentions to the developers !!!

To meet the requests of many developers, we have added to CreaBar two DLL that can be connected to other programs.

CompuArt encourages you to use its programs with other ones produced by other outside parties: remember that the professional versions are not entirely or partially freely distributable.

You can use the DLL in your applications, but you must have the registered (professional) version of the program CreaBar: you can receive it directly from CompuArt or, by authority received, you can sell it.

CompuArt makes particular discounts to the developers that want to use its programs. Contact us !

GBCODE.DLL and GBCODE32.DLL libraries

CreaBar has two dynamic link libraries that allows to create a bar code from the other applications inside:

- GBCODE.DLL for 16 bit applications
- GBCODE32.DLL for 32 bit applications

The Setup program purposes the CreaBar installation in the directory C:\CBAR20, while the libraries and the file CBAR20.RG are installed in the directory Windows in order to be available to all the applications. During an eventual disinstallation, the program will ask you a confirm to remove the file .RG, the libraries and the directory containing the program (CBAR20); the CreaBar disinstallation will be completed if you confirm this request: in this way no tracks of the program will remain in the hard disk.

The available procedures in GBCODE/GBCODE32.DLL are the following:

```
procedure GetVersionInfo(var Vers.IsSw:smallint)
procedure GetClpMetaCode(FileCode,DataCode:Pchar;var Err:smallint;var MetaH:THandle)
procedure GetTestString(FileCode,S:Pchar;Count:smallint)
procedure PutCodeInClipboard(FileCode,DataCode:Pchar;var Err:smallint)
```

The ordinal declarations are the following:

```
GetVersionInfo      index 1
GetClpMetaCode     index 2
GetTestString      index 3
PutCodeInClipboard index 4
```

[Example of link to Visual Basic 3.0](#)

[Example of link to Corel Draw 7 Script Editor](#)

[Example of dynamic link to Borland Pascal or Delphi](#)

[Example of use in Pascal of the function GetClpMetaCode](#)

GetVersionInfo

procedure GetVersionInfo(var Vers,IsSw:smallint)

This procedure writes the information about the GBCode version. It requests two entire values (signed word) where the data are written. They must be passed by a variable reference.ì

The version number multiply by 100 is written in **Vers**. For example the version 2.0 gives the value 200. The value 1 will be written in **IsSw** if its a Shareware version of GBCode, while the value 0 will be written if its a Professional version.

Other procedures

GetClpMetaCode

procedure GetClpMetaCode(FileCode,DataCode:Pchar;var Err:smallint;var MetaH:THandle)

This procedure allows you to obtain, from GBCode, a memory form that contains the bar code metafile. **FileCode** and **DataCode** are two ASCII string pointers, i.e., they are variable length strings that end with the character 0. They must be passed by value.

FileCode must contain, in input, the name of the file *.BRC; this file contains the characteristics for the creation of the bar code. You must use the program CreaBar to create these files. The file name must contain also the path.

DataCode must contain, in input, the datum (as a string) that must be codified in the bar code. The datum must have a valid form according to the code to create, for example 12 numeric ciphers for EAN13, 7 for EAN8, numbers, letters and the following symbols "-" "." ";" "\$" "/" "+" "%" for Code39. The data must be the same ones that CreaBar uses in its input line for every type of code.

Err restores an error number or a zero. It's an integer type (signed word) and it must be passed by a variable reference.

MetaH pointer to MetaFilePict structure contains the metafile and its information.

If the creation has reached a bad result, the value is equal to 0. It must be passed by a variable reference.

Its the form used by Windows to the interchange of the metafiles throught the Clipboard: its form is

```
typedef struct METAFILEPICT {  
    int mm;  
    int xExt;  
    int yExt;  
    HMETAFILE hMF;  
} METAFILEPICT;
```

To go deep into this thema (how you can explain and use this structure), we advise you to consult specific texts (as "Programming Windows" - Charles Petzold).

The procedure that uses the structure must release its associated memory at the end of the job.

Other procedures

GetTestString

procedure GetTestString(FileCode,S:PChar;Count:smallint)

It gives back a valid string for the type of code specified in FileCode. It can be used to show a default value in the input line where you want to write the data to codify.

FileCode and **S** are two ASCII string pointers, i.e., they are variable length strings that end with the character 0. They must be passed by value.

FileCode must contain, in input, the name of the file *.BRC; this file contains the characteristics for the creation of the bar code. You must use the program CreaBar to create these files. The file name must contain also the path.

S must point to a buffer where the text string will be copied (generally 20 characters at most).

Count is equivalent to the maximum number of characters that GetTestString copies in S. Count specifies the number of bytes of the buffer that will receive the string. It's an integer (signed word) and it must be passed by value.

Other procedures

PutCodeInClipboard

procedure PutCodeInClipboard(FileCode,DataCode:PChar;var Err:smallint)

Its equivalent to GetClpMetaCode, but it puts the created structure directly in the Clipboard and doesn't restore the MetafilePict Handle. This function is very useful for everyone that uses macro languages or high level languages; these languages make the right management of the MetafilePict structure or the correlative memory management impossible.

It's possible, using this procedure, create a bar code, put it in the Clipboard and paste it, with the command Paste, in the document.

FileCode and **DataCode** are two ASCII string pointers, i.e., they are variable length strings that end with the character 0. They must be passed by value.

FileCode must contain, in input, the name of the file *.BRC; this file contains the characteristics for the creation of the bar code. You must use the program CreaBar to create these files. The file name must contain also the path.

DataCode must contain, in input, the datum (as a string) that must be codified in the bar code. The datum must have a valid form according to the code to create, for example 12 numeric ciphers for EAN13, seven for EAN8, numbers, letters and the following symbols "-" "." "," "\$" "/" "+" "%" for Code39. The data must be the same ones that CreaBar uses in its input line for every type of code.

Err restores an error number or a zero. It's an integer type (signed word) and it must be passed by a variable reference.

Other procedures

Errors restore from GBCode

The variable Err restored from GetClpMetaCode and PutCodeInClipboard can put on the following values:

0 no errors

1 GBCode cannot find the specified configuration BRC file

2 error in the code creation. The values put in the configuration file dont allow you to create a bar code geometrically right

3 error in the string to convert. The number or the quality of the specified string is not in conformity with the characteristics of the bar code to create.

If Err is different from zero, the restored memory handle has not meaning or nothing is copied in the clipboard.

Other procedures

Example of GBCODE.DLL link to Visual Basic 3.0

(In General - Declaration)

```
Declare Function PutCodeInClipboard Lib "gbcodes.dll" (ByVal F As String, ByVal D As String, E As Integer) As Integer
Dim filecode As String
Dim datacode As String
Dim erro As Integer
Dim error As Integer
```

```
Sub Form_Click ()
    error = 0
    filecode = "code39.brc"
    datacode = "F41-422410001"
    clipboard.Clear
    error = PutCodeInClipboard (filecode, datacode, erro)
    picture1 = clipboard.GetData ()
    text1.Text = Str$(error)
End Sub
```

Other procedures

Example of program written using Corel Draw 7 Script Editor

```
REM *****
REM * Global Data *
REM *****

DECLARE SUB GetVersionInfo LIB "GBCODE32.DLL" (BYREF Vers% AS
INTEGER,BYREF IsSw% AS INTEGER) ALIAS "GetVersionInfo"
DECLARE SUB PutCodeInClipboard LIB "GBCODE32.DLL" (BYVAL FileCode$ AS
STRING, BYVAL DataCode$ AS STRING, BYREF Err% AS INTEGER) ALIAS
"PutCodeInClipboard"

REM *****
REM * Main Dialog *
REM *****
BEGIN DIALOG OBJECT MainDialog 200, 112, "Test of link to CreaBar", SUB
MainDialogSub
    TEXT 7, 9, 42, 9, .Text3, "File CreaBar:"
    TEXTBOX 52, 7, 79, 13, .CBarFile
    PUSHBUTTON 135, 6, 60, 14, .Browse, "Browse"

TEXT 23, 33, 26, 8, .Text2, "Code:"
    TEXTBOX 52, 31, 103, 13, .Code
    PUSHBUTTON 62, 53, 80, 14, .PushButton2, "Copy in the clipboard"
    TEXT 4, 73, 190, 10, .Info, "Do not connected to CreaBar"
    TEXT 4, 84, 193, 22, .Report, "Text6"
END DIALOG

'Initialize the styles for the controls
MainDialog.SetStyle STYLE_MINIMZERESIZE

'Execute the dialog
DIALOG MainDialog

REM *****
REM * MainDialogSub: *
REM *****
SUB MainDialogSub(BYVAL ControlID%, BYVAL Event%)
    DIM FileName AS STRING
    DIM DialogWidth AS INTEGER
    DIM DialogHeight AS INTEGER
    DIM ImageHeight AS INTEGER
    DIM ImageWidth AS INTEGER

DIM Vers% AS INTEGER
    DIM IsSw% AS INTEGER
    DIM Info$ AS STRING
    DIM Num$ AS STRING
    DIM FileCode$ AS STRING
    DIM DataCode$ AS STRING
    DIM Err% AS INTEGER
    WITH MainDialog
        IF Event%=0 THEN
            CALL GetVersionInfo Vers%, IsSw%
```

```

        Num$=STR(Vers%)
        Info$="Collegato a CreaBar "+Num$
        IF IsSW=0 THEN
            Info$=Info$+" Registered"
        ELSE
            Info$=Info$+" Shareware"
        ENDIF
        .Info.SetText Info$
BEGIN DIALOG Dialog1 200, 100, "Corel SCRIPT Dialog"
END DIALOG
        .Report.SetText ""
    ENDIF
    IF ControlID%=3 THEN
        IF Event% = EVENT_MOUSE_CLICK THEN
            FileName$ = GETFILEBOX("Files CreaBar (*.brc)*.brc")

IF FileName$ <> "" THEN
                .CBarFile.SetText FileName$
            ENDIF
        ENDIF
    ENDIF
    IF ControlID%=6 THEN
        IF Event%=EVENT_MOUSE_CLICK THEN
            FileCode$=.CBarFile.GetText()
            DataCode$=.Code.GetText()
            CALL PutCodeInClipboard FileCode$, DataCode$, Err%
            SELECT CASE Err%
                CASE 0
                    .Report.SetText "The code is copied in the
Clipboard"
                CASE 1
                    .Report.SetText "The code configuration file does not
exist "+ .CBarFile.GetText()
                CASE 2
                    .Report.SetText "Error in the creation
of the code"
                CASE 3
                    .Report.SetText "Error in the string to convert"
            END SELECT
        ENDIF
    ENDIF
END WITH
END SUB

```

Other procedures

Example of dynamic link in Pascal

Unit LinkCBar;

Interface

Uses Objects, OWindows, WinTypes, WinProcs;

(* procedures dynamically linked to the DLL GBCode of CreaBar *)

type

```
TCreaBarGetVersionInfo=procedure(var Vers,IsSw:integer);
TCreaBarGetTestString =procedure(FileCode,S:PChar;Count:integer);
TCreaBarGetClpMetaCode=procedure(FileCode,DataCode:PChar;var Err:integer;var
MetaH:THandle);
```

var

```
CreaBarGetVersionInfo:TCreaBarGetVersionInfo;
CreaBarGetTestString:TCreaBarGetTestString;
CreaBarGetClpMetaCode:TCreaBarGetClpMetaCode;
```

var

```
HGBCode:THandle; (* DLL GBCode handle *)
```

```
CreaBarIsLinked:boolean; (* true if GBCode.dll is linked *)
```

```
CreaBarVersion:integer; (* version of the linked CreaBar *)
```

```
CreaBarShareware:boolean;(* true if the linked CreaBar is shareware *)
```

```
function ConnectToCreaBar:boolean;
```

Implementation

```
function ConnectToCreaBar:boolean;
```

var

```
ErrMode:word;
Version,IsShareware:integer;
begin      (* ConnectToCreaBar *)
CreaBarIsLinked:=false;
(* do not write the message if GBCode does not exist *)
ErrMode:=SetErrorMode(sem_NoOpenFileErrorBox);
HGBCode:=LoadLibrary('GBCode.dll');
SetErrorMode(ErrMode);
if HGBCode<HInstance_Error
then HGBCode:=0
else
begin      (* GBCode OK *)
  @CreaBarGetVersionInfo:=GetProcAddress(HGBCode,'GetVersionInfo');

  if @CreaBarGetVersionInfo<>nil
  then
  begin
    CreaBarGetVersionInfo(Version,IsShareware);
```



```

    CreaBarVersion:=Version;
    CreaBarShareware:=IsShareware=1;
    CreaBarIsLinked:=true;
    @CreaBarGetTestString:=GetProcAddress(HGBCode,'GetTestString');
    @CreaBarGetClpMetaCode:=GetProcAddress(HGBCode,'GetClpMetaCode');
end
else
begin
    FreeLibrary(HGBCode);
    HGBCode:=0;
end;
end;
ConnectToCreaBar:=CreaBarIsLinked;
end;      (* ConnectToCreabar *)

```

(* exit code of the unit. It is used to free the memory occupied by the module *)

```

var
ExitSave:Pointer;

procedure ExitFromUnit; far;
begin      (* ExitFromUnit *)
    ExitProc:=ExitSave;
    if CreaBarIsLinked
    then
    if HGBCode<>0 then FreeLibrary(HGBCode);
end;      (* ExitFromUnit *)

```

```

initialization
(* initialize the parameters for CreaBar *)
CreaBarIsLinked:=false;
HGBCode:=0;
(* initialize the exit code *)
ExitSave:=ExitProc;
ExitProc:=@ExitFromUnit;
end.

```

Other procedures

Example of use in Pascal of GetClpMetaCode

This is the procedure that Label Laboratory uses to draw bar codes by CreaBar.

```
procedure TBaseFieldBarCode.DrawMetafile(DC:HDC;R:TRect;var DrawErr:integer);
var
  HMetaFile,MetaH:THandle;
  Err:integer;
  MetaPict:PMetaFilePict;
  DX,DY:integer;
  Pt:array[1..2] of TPoint;
  RX,RY:real;
  Rgn:HRgn;
  S:array[0..300] of char;
  Ret:array[0..2] of char;
  T:PChar;
  I:integer;
begin
  (* TBaseFieldBarCode.DrawMetafile *)
  if not CreaBarIsLinked
  then
    begin
      DrawErr:=drawErr_CBarUnLink;
      Exit;
    end;
  if FileName=nil then Exit;
  if StrLen(FileName)=0 then Exit;

  if TextField=nil then Exit;
  if StrLen(TextField)=0 then Exit;

  (* the text must be in the first row *)
  StrLCopy(S,TextField,299);
  Ret[0]:=#13;
  Ret[1]:=#0;
  T:=StrPos(S,Ret);
  if T<>nil then T[0]:=#0;
  if StrLen(S)=0 then Exit;

  CreaBarGetClpMetaCode(FileName,S,Err,MetaH);
  if Err<>0
  then
    begin
      if Err=1
      then DrawErr:=drawErr_NoFile
      else DrawErr:=drawErr_BadCod;
    end
  else
  if MetaH<>0
  then
    begin
      SaveDC(DC);

      (* set the mapping mode of the rectangle in device coordinates *)
      with R do
```

```

begin
  Pt[1].X:=Left;
  Pt[1].Y:=Top;
  Pt[2].X:=Right;
  Pt[2].Y:=Bottom;
end;
LpToDp(DC,Pt,2);

(* create a clipping region *)
Rgn:=CreateRectRgn(Pt[1].X,Pt[1].Y,Pt[2].X,Pt[2].Y);
SelectClipRgn(DC,Rgn);

MetaPict:=GlobalLock(MetaH);
with MetaPict^ do
begin

  SetMapMode(DC,mm_Anisotropic);
  RX:=xExt/10;
  RY:=yExt/10;
  DX:=Round((RX*GetDeviceCaps(DC,LogPixelsX)*ZoomFact)/254.0);
  DY:=Round((RY*GetDeviceCaps(DC,LogPixelsY)*ZoomFact)/254.0);

  SetViewPortOrg(DC,Pt[1].X,Pt[1].Y);
  SetViewPortExt(DC,DX,DY);
  HMetaFile:= hMF;

  if (DX<(Pt[2].X-Pt[1].X)) or (DY<(Pt[2].Y-Pt[1].Y))
  then DrawErr:=drawErr_Overflow;
end;
GlobalUnLock(MetaH);

PlayMetaFile(DC,HMetaFile);

GlobalFree(HMetaFile);
GlobalFree(MetaH);

SelectClipRgnToNull(DC);
DeleteObject(Rgn);

RestoreDC(DC,-1);
end;
end;      (* TBaseFieldBarCode.DrawMetafile *)

```

Other procedures

Creation of labels

CreaBar is very useful to insert bar codes in every type of document, but if you want to create labels using the data of a database, you should use a specific program. *CompuArt*, according to the requests of its clients, has realized the program **Label Laboratory**. It has a complete management of databases in dBASE III form and a graphic editor that allows you to draw every type of label. Ask us information about it to the usual address.

Summary

Registration

We have tried to make the registration easier: in fact, inside the program, there is a form where you will put your data. This form is available from the option "Registration" from the menu "File".

In this way we will insert your data in our clients database; so you will have the advantage to be informed about the programs upgrades and all the news.

When you have inserted all the data, you can send the order form by e-mail or print and send it:

by fax to the following numbers: **+39 10 6530179 - +39 10 6513199**

by mail to the following address:

COMPUART

C.P. 5541

16158 GENOVA VOLTRI

ITALY

If there is any problem during the registration by e-mail, you can contact us to the following address:

info@compuart.it

The program registration costs **Italian Lire 46.500 (about 27\$ - US Dollars)**.

We only accept the advance payment by credit card (VISA, American Express, MasterCard, EuroCard, Electron), cash enclosed or postal order payable to:

COMPUART - C.P. 5541 - 16158 GENOVA VOLTRI - ITALY.

When you have printed the order or sent it by e-mail, the program waits an activation code that you will receive from CompuArt by e-mail, fax or mail.

In this pre-registration phase, the program, every time that it is used, asks the activation code that allows you to finish the registration and activate all the program functions.

The file CBAR20.RG is created in the Windows directory: it contains all the registration data.

Remember to do a backup copy of this file on floppy disk: if you have to reinstall the program, install the shareware version and copy this file in the Windows directory.

N.B.: When you are in the pre-registration phase (when the program shows, at the start, the code request message), you **MUST NOT** enter in the registration window and change its content. It's because a new product code, that has a different activation code, is created: in this way the activation code you are waiting from CompuArt is useless.

You can change the data as you like if you have not yet request the program registration, i.e., if the printed order form has not been yet sent by fax or surface mail, or if the forwarding by e-mail doesn't get a good result.

Summary

Differences between the shareware and the professional version

The two versions are perfectly functional, but the shareware version has the codify system only for the following codes: EAN 8 and EAN 13, so you can value directly the program. A fixed string is created for the other codes but you cannot modify it.

There are not limits of time for the registration. At the moment of the registration, we will send you an activation code that allows you to transform the shareware version in the professional and use all the program functions.

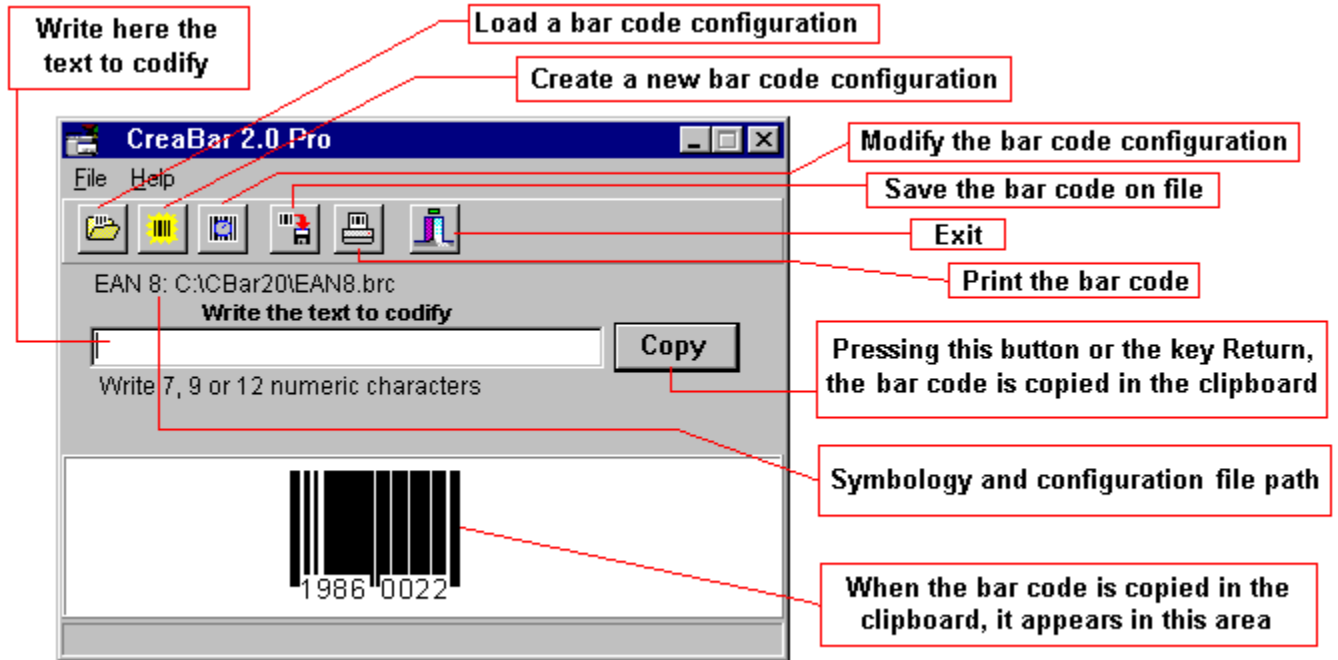
Summary

Bar codes generated by the program

Registration

Rapid show to the program

The main program's aim is to create bar codes, copy them in the Windows clipboard and from here paste them in another application. Creabar look has been studied in order to take up the little space possible on the screen, because it is used together with other applications. The following picture shows the use of the window areas.



Summary

List of EAN prefixes and National Associations for the distribution of the product code

Prefixes	National Associations
00 - 09	UCC (U.S.A. + Canada)
20 - 29	In-store numbers
30 - 37	GENCOD (France)
40 - 43	CCG (Germany)
460 - 469	USSR CCI (USSR)
471	ANC of ROC (Taiwan)
489	HKANA (Hong Kong)
49	Distribution Code Center (Japan)
50	ANA Ltd. (UK) and ANA of Ireland
520	HELLICAN (Greece)
5299	Cyprus Chamber of Commerce and Industry (Cyprus)
54	ICODIF (Belgium + Grand Duchy of Luxemburg)
560	CODIPOR (Portugal)
569	Iceland EAN-Committee (Iceland)
57	Dansk Varekode Administration (Denmark)
590	Polish Chamber of Foreign Trade (Poland)
599	Hungarian Chamber of Commerce (Hungary)
600 - 601	SAANA (South Africa)
64	Central Chamber of Commerce (Finland)
70	Norsk Varekodeforening (Norway)
729	Israel Coding Association (Israel)
73	Swedish EAN Committee (Sweden)
750	AMECOP (Mexico)
759	CIP (Venezuela)
76	SACV (Switzerland)
770	IAC (Colombia)
773	CUNA (Uruguay)
775	APC (Peru)
779	CODIGO (Argentina)
780	CNC - DEPCO (Chile)
789	ABAC (Brazil)
80 - 83	INDICOD (Italy)
84	AECOC (Spain)
850	Camera di Commercio de la Republica de Cuba (Cuba)
859	Czechoslovak CCI (Czechoslovakia)
860	JANA (Jugoslavia)
869	Union of Chambers of Commerce of Turkey (Turkey)
87	STICHTING UAC (Netherlands)
880	KORREA CCI (South-Korea)
885	Thai Product Numbering Association Ltd. (Thailand)
888	SANC (Singapore)
90 - 91	EAN - AUSTRIA (Austria)
93	APNA Ltd. (Australia)
94	NZPNA Ltd. (New Zealand)
955	Malaysian Article Numbering Council (Malaysia)
959	PNGPNA (Papua New Guinea)
977	Periodicals (ISSN)
978 - 979	Book (ISBN)
98 - 99	Coupon numbers

Bar codes EAN

