Printing Barcodes

By: Thomas Maul, General Manager, 4D Germany
Technical Note 05-08

Summary

Bar codes are widely used for automatic indentification on products, shipment papers, etc.

This technical note describes a 4D component (source code included) allowing to print a set of different bar codes with a single command.
There is no need to understand how the concept works. See Tech Note 02-20 for background information on how bar codes work and how they are created. The component supports calculating check sums and converting UPC-A to UPC-E.
The technical note is divided into two main sections: Bar code types and using the component.

The Bar code component

Because this component is divided between pattern (black/white bars) calculation and producing bar codes it can be enhanced for additional bar code types.
The tech note is provided with an example using the component and the source code of the component.
The component makes it very easy to use bar codes, it has only two (visible) methods, for most cases using one method is all a developer needs. Still the source code is provided if a developer wants to learn how it is done or if there is a need for modification.
The component is created using 4D Insider 2004.1 so it is only useable with 4D 2004.1 or newer. For older versions the source code can be used to install the component directly or to create a component with an older version of 4D Insider.

Supported Bar Codes

- Industrial 2 of 5
- Interleaved 2 of 5
- Code 39
- Code 128 (A, B, C)
- EAN 8 and EAN13
- UPC-A and UPC-E
- 2 and 5 digit supplements for EAN and UPC

Industrial 2 of 5

Supported characters: only digits (0-9)
Max number of characters: -
Check sum: optional
Compactness: low

Interleaved 2 of 5

Supported characters: even number of digits (0-9)
Max number of characters: - count must be even
Check sum: optional
Compactness: medium
Used by: USPS Special Services (US Mail), SCC-14 - EAN-14 (shipping papers)
Special: pairs digits are "interleaved", first builds the black bars, second the white bars, to reduce width

Code 39

Supported characters: 0-9, A-Z, "-. $/+%" (note: A-Z does not include diacritic characters like umlauts)
Max number of characters: 30
Check sum: optional
Compactness: medium
Used by: LOGMARS (United States Department of Defense standard), 4D product boxes

Code 128

Code 128 has 3 subtype, 128A, 128B and 128C.
128A supports control characters (char 0-31), 128B supports lowercase characters and 128C is reduced to digits only.
The code supports switching the subtype in the middle of a code, by example to start with 128B and then switch to 128C to be more efficient for digits.
Switching is not supported from this component.

Max number of characters: -
Check sum: mandatory (printing check sum as text is optional)
Compactness: medium
Used by: USPS Special Services (US Mail), SSCC18 - EAN128 - UCC128 (shipping papers)

Code 128A

Supported characters: 0-9; A-Z;!\"#$0% & '()*+0-./: ;<=>?@[\\]^_"+" control char(00-31)
(note: A-Z does not include diacritic characters like umlauts)

(example code is "4DDE700TEST", ">" is the calculated checksum)

Code 128B

Supported characters: 0-9; A-Z;a-z;!\"#$0% & '()*+0-./: ;<=>?@[\\]^_{}~`
(note: A-Z or a-z does not include diacritic characters like umlauts)

(example code is "4DDE700Test", "`" is the calculated checksum)

Code 128C

Supported characters: even number of digits (0-9)

(example code is "0123456789", "73" is the calculated checksum)

EAN 8, EAN 13, UPC-A, UPC-A, ISBN, ISSN

Supported characters: 0-9
Max number of characters: 7-13
Check sum: mandatory, printing in text mandatory
compactness: high
Used by: UPC, EAN, ISBN, ISSN, Bookland, consumer products worldwide

This bar code family is seen daily. It is used in super markets, book stores and all over the world at cash registers. It is the most complex to read and print.

UPC (Universal Product Code) was introduced April 3 in the USA and Canada and is used only in these countries. Code numbers are strictly restricted, they are controlled from Uniform Code Council (UCC). The code is divided into a manufacturer code and product number to give a worldwide unique number.

UPC-A is a 12 digit (11 + check sum), UPC-E is a compressed code with only 7 digits (+ checksum, used for small articles with reduced available space. Only some Codes are possible to convert to UPC-E, these ranges are specially assigned from the UCC.
The component supports converting UPC-A to UPC-E.

EAN (European Article Number) is the European counterpart of UPC, it is based on UPC. EAN is usually 13 (12+1) digit, so one more than UPC. There is also a reduced set (EAN 8 with 7 + 1 digit) for small articles, but there is no way to calculate (compress) a EAN 13 to 8, this are specially designed numbers. Also the checksum calculation from EAN 8 is different.

Because EAN is based on UPC, all EAN-compatible scanners can read UPC - they handle it like an EAN 13 code with a leading 0. The first 2 digits of an EAN 13 define the country which manage the number ranges (for some smaller countries it has subranges which are defined with the first 3 digits, like 539 = Ireland). The range 00-13 is reserved for USA and Canada, this contains all numbers of the UPC-E range.

The numbers are worldwide unique, numbers cannot be simply used, they have to be assigned/registered from the local UCC (for U.S: www.uc-council.org, international: http://www.ean-ucc.org/). For super markets and internal number systems the EAN codes starting with "2" are reserved, the remaining numbers can be used to code the product and price/weight.

ISBN (International Standard Book Number - code 978), ISSN (International Standard Serial Number - code 977) and ISMN (International Standard Music Number - code 979) are based on EAN 13.

To print a ISBN number as Barcode it needs to be converted, which can be done very simply.

Let's use the ISBN number of the German edition of Jumpstart 4D by Steve Hussey as example, it is:
ISBN: 0-9712895-2-2
we add the "country code" of ISBN, which is 978 before:
EAN: 978 097128952
Note that the last digit of the ISBN, which is the check sum, is removed, it will be replaced with the EAN 13 check sum. The component contains a method to calculate this check sum (so you can store it as part of the article number), it also automatically calculates the checksum for printing if is not contained.

Books and magazines often use a 2 or 5 digit supplemental bar code, see special section below.

EAN in United States - New rules starting 1-1-2005

While all EAN scanners was able to read UPC (and Bar code printing applications are able to print UPC-A by simply adding a leading "0"), older UPC scanners was not able to read EAN and most printing software was unable to manage numbers with more than 12 digits.

The solution used until 2004 was to give to European and Asian companies, selling products to U.S. and Canada, a company prefix to be used for UPC bar codes. Beside that, these companies needed to print two different bar codes on their products (which both looked very similar) - it is obvious that is was only a question of time until the UCC would run out of numbers.

New solution - Sunrise 2005

As of January 1, 2005, all POS (point of sale) applications to be used in U.S or Canada (and these are scanners and software) must be able to handle EAN-8, EAN-13 in addition to UPC-E and UPC-A codes.

European companies are no longer given US company prefixes, they uses "their" European EAN code on products sold in the U.S.

Changes in UPC codes in the U.S
U.S. based companies are given company prefixes which were previously "reserved", like codes starting with 1, 8 or 9.

Some POS software products stored only 11 digits (some without checksum - some without the leading 0, previously UPC-E was always starting with 0).
Some POS software products even stored only 10 digits (without checksum and without leading 0).

Global Trade Item Number (GTIN)

Starting January 1, 2005, software products are supposed to store a 14 digit number.
Scanners are supposed to automatically provide 14 digit numbers.
This is done by adding leading zeros to reach 14 digits (EAN 8 gets 6 leading zeros).
Scanners sold in the last couple of years in the U.S. are able to read EAN but usually give only the "real" number. It may be wise for a software product to recognize these and automatically add the leading zeros.

To print a GTIN number as bar code the leading zeros have to be removed first. The "real" numbers remains unchanged, so no new numbers have to be given to the manufacturers or to be printed.

The component automatically removes leading zeros from GTIN numbers.

EAN 13

Used on product boxes in Europe, and, starting in 2005, also in the U.S and Canada. Optically easy to recognize, the start, middle and end bars are two longer thin lines. To read those codes, it is not necessary to have them in a different size, it is only done to visually recognize it as EAN code. The 13-digit number is divided as follows: first number on the left of the code, then 6 digits for each half.

On the right side of the code there may be a 2 or 5 digit supplemental code, the example above is the ISBN code for the US issue of Jumpstart 4D with the suggested retail price of US $29.99.

EAN 8

Used on product boxes in Europe, starting 2005 also in U.S and Canada. Optically easy to recognize, the start, middle and end bars are two longer thin lines. . To read those codes, it is not necessary to have them in a different size, it is only done to visually recognize it as EAN code. The 8-digit number is divided as follows, first half in the left of the code, remaining 4 digits in right part.

On the right side may be a 2 or 5 digit supplemental code

UPC-A

Used on product boxes only in U.S and Canada. Optically easy to recognize, the start, middle and end bars are two longer thin lines. To read those codes, it is not necessary to have them with a different size, it is only done to visually recognize them as an UPC code. The 12-digit number is divided as follows: first number on the left of the code, then 4 digits in each half, last digit on the right side of the code.

On the right side of the code may be a 2 or 5 digit supplemental code, the example above shows the number 12. It should be used only for magazines, newspapers and other periodicals, showing the issue number. This permits having a single product code for a magazine and still be able to detect the issue number.

UPC-E

Used on product boxes only in U.S and Canada. Optically easy to recognize, the start, middle and end bars are two longer thin lines. To read this code, it is not necessary to have those bars in a different size, it is only done to visually recognize it as UPC code. The 8-digit number is divided as follows: first number on the left of the code, than 6 digits inside, last digit on the right side of the code.
The code contains only 8 from the original 12 numbers, 4 are suppressed. The code can be converted loss-less from UPC-E to -A and back, which

means only a special range of numbers can be used to create UPC-E. For example the code 0-42100-0052-6 is printed as UPC-E as 0-425261-4. Companies needing UPC-E codes have to request a manufacturer code in a special area. There are manufacturer codes available which allows to encode 5, 10, 99 or 999 products.

On the right side may be a 2 or 5 digit supplemental code

Supplemental codes for EAN and UPC

On the right side of the code may be a 2 or 5 digit supplemental code.
A 2 digit number should be used only for magazines, newspapers and other periodicals, showing the issue number. This allows to have a single product code for a magazine and still be able to detect the issue number.

5 digit numbers are for price information. In U.S. the first digit expresses the currency, "5" indicates US$, the following 4 digits are the price itself. Example: "51195" indicates a suggested retail price of US$ 11.95. This limits the range for price for US books to 99.99...

In UK a book using "01199" will be read as British Pounds 11.99, while in Germany as Euro 11.99, showing that it cannot be used as safe currency indicator. German book stores selling imported books from UK or US usually put their own printed labels on top of the original bar code, using Euro values.

Codes starting with 9 have special meanings:
90000 no suggested retail price
99991 complimentary copy - not for resale
99990 Used by National Association of College Stores to mark used books
900001-98999 To be used for internal purpose for publishers.

Supplemental codes are printed with the printed values on top of the code with a shorter length than the EAN/UPC code.

Using the component - producing bar codes

The component allows you to create a picture, representing the barcode, with a single command.

Barcode_Create (barcodetype; code; createchecksum; showchecksum; printcode; {chartarea}) -> Barcodepicture

Parameter	Type	Description
barcodetype	Text	Type of Barcode
code	Text	Code/Number to produce
createchecksum	Boolean	Create and add a checksum
showchecksum	Boolean	Show the checksum in printed clear text
printcode	Boolean	Print the code as clear text beyond
chartarea	Longint	Reference to an existing 4D Chart area
Barcodepicture	Picture	Created picture/barcode

Desription
The command Barcode_Create creates a barcode as a picture.

The parameter barcodetype indicates the type of the barcode, it must be one of the following supported types:

- Industrial 2 of 5
- Interleaved 2 of 5
- Code39
- Code128A
- Code128B
- Code128C
- EAN8
- EAN13
- UPC-A
- UPC-E
- Supplemental2
- Supplemental5

The parameter code specifies the code to create. Depending on *barcodetype* there are several rules for the allowed content or length. See the description of the bar code types for these rules. If *code* contains unsupported characters or an invalid length the returned picture is empty.

Createchecksum allows you to enable a check sum calculation, note that this is mandatory for some bar code types.
Showchecksum allows you to hide the check sum in the printed clear text beyond the bar code. Showing the check sum is mandatory for EAN and UPC.
PrintCode allows you to disable the printing of clear text beyond the code. Printing the text is mandatory for EAN and UPC.

The optional parameter *chartarea* allows you to specify an already existing 4D Chart area to use for picture creation. If the parameter is not passed, the routine creates a new 4D Chart area and disposes of it automatically. If there are many bar codes to create it speeds up the process to create it only once and reuse the area.

Creating 100 EAN codes (compiled mode on an AMD 2400 CPU with Windows XP) needs 3200 ticks (this is 5.5 seconds) if the area needs to be re-created each time and 120 ticks (2 seconds) if it is re-used.
It is important to clear the area (using CT New Document) after each bar code to reset the area!

Optional settings

The Width, Height and Font are automatically set - they can be overwritten. There are 6 variables which can be set to change the behavior:

Barcode_Width	1	Width of a single bar
Barcode_Height	40 (25)	Height of the code (automatically reduced for additional)
Barcode_Add	3	Additional height for start/middle/end bars in EAN/UPC
Barcode_Font	Arial	Font to print clear text
Barcode_FontSize	9	Font Size
Barcode_FontOffset	5	Distance from Bar code to clear text

If *Barcode_Width* is 0 the method will set the values automatically, else it use the preset values. In this case all values must be set!

To improve the readability with inkjet printers the routine automatically creates a 300 dpi bitmap. Testing showed that the readability improved drastically compared to 72dpi vector pictures.
This is done by creating the picture 4 times bigger (with 4-pixel width of a single bar, using a 36-point font and so on).
After creation the picture is converted to a bitmap by:
$picture := $picture | $picture
and then scaled down:
$picture := $picture * 0.25

This calculation is only done in automatic mode - if *Barcode_Width* equals 0. If variables are preset, the picture is returned like it is in vector mode. We recommend to always scale, values of 3 or 4 for *scaling* have given good results in testing.

Example

```
$barcode:=Barcode_Create    ("Code128B";"4DDE700";True;True;True)


barchart:=CT New offscreen area
$scale:=4 `  create the picture bigger, increase readability for inkjet printers
Barcode_Height:=40*$scale
Barcode_Width:=1*$scale
Barcode_Add:=3*$scale
```

```
Barcode_Font:="Arial"
Barcode_FontSize:=9*$scale
Barcode_FontOffset:=5*$scale
$barcode:=Barcode_Create    ("Code128B";"4DDE700";True;True;True;barchart)
$barcode:=$barcode | $barcode `  convert to bitmap
$barcode:=$barcode*(1/$scale) `  scale back to original size
CT NEW DOCUMENT  (barchart)
$barcode2:=Barcode_Create    ("Code128B";"4DDE800";True;True;True;barchart)
$barcode2:=$barcode2 | $barcode2 `  convert to bitmap
$barcode2:=$barcode2*(1/$scale) `  scale back to original size
CT DELETE OFFSCREEN AREA  (barchart)
```

Barcode_Calc_Checksum (barcodetype; code) -> Checksum

Parameter	Type	Description
barcodetype	Text	Type of Barcode
code	Text	Code/Number to produce
Checksum	Text	Created checksum

Desription

The command Barcode_Calc_Cecksum creates a checksum for the code passed, in compliance with the rules for the bar code type.

The parameter *barcodetype* indicates the type of the barcode, it must be one of the supported types.
The parameter code specifies the code to create. Depending on *barcodetype* there are several rules for the allowed content or length. See the description of the bar code types for these rules. If *code* contains unsupported characters or an invalid length the returned check sum is "?".

Checksum is 1 or 2 character in the supported character range (depending of *barcodetype*). For Code128C it will be a 2-character string using digits (0-9), else a one character string.

Using this method is not mandatory, check sums can also be automatically calculated during creating the bar code.

This method helps if it is needed to store the check sums in an inventory system or to check if an entered number is correct.

Examples

```
$number:="12345678"

$checksum:=Barcode_Calc_Checksum    ("EAN8";Substring($number;1;7))
If ($checksum#Substring($number;8;1))
        ALERT("The  EAN8 number "+$number+" is incorrect (check sum invalid)")
End if
```

Addendum: Barcodes using fonts

It is also possible to print bar codes using fonts, and many people use that solution.
The main disadvantage is that the fonts need to be installed on the computers of all users.
This makes user support more difficult.
Usually these fonts are commercial, but there are also free fonts available.
Here an URL to PostScript Type 1 fonts available for free (GNU license), allowing Code 128, code 39, Interleaved 2 of and UPC A, E and EAN 8 and 13.
http://user.it.uu.se/~jan/barfonts/

Links

Barcodes:
http://www.barcodeisland.com/