







ELFRING FONTS

THE RETAIL BAR CODE SOLUTION

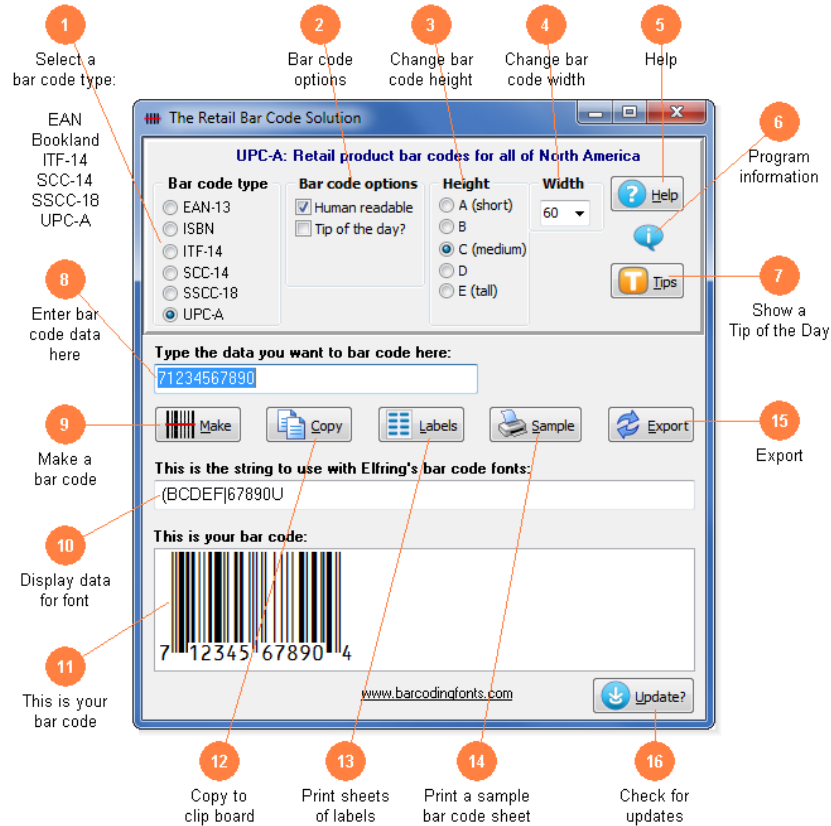
This package includes every kind of bar code needed to produce retail product and shipping bar codes. These bar code types include EAN-13, ISBN (Bookland), ITF-14, SCC-14, SSCC-18, and UPC-A. All of these bar codes are quite complicated. You can not simply type in your bar code number and display it using a bar code font. You must use our Retail.exe utility program or our Visual Basic macros in Excel or Access to build these bar codes for you.

Bar Code	Bar Code Description
	EAN-13 bar codes are used to identify retail products in all countries except the US and Canada. You must purchase your product number assignment from gs1.org or other sources to make this kind of bar code.
	ISBN or Bookland bar codes turn your ISBN number in a book bar code. The bar code identifies the book and a second optional bar code specifies the book price and currency.
	The ITF-14 bar code is based on your UPC or EAN assignment and is used on the outside of boxes, cases, and pallets to identify the product within.
	The SCC-14 (Shipping Container Code) is based on your UPC or EAN manufacturer's number assignment and is used on fixed content shipping containers.
	The SSCC-18 (Serial Shipping Container Code) is based on your UPC or EAN manufacturer's number assignment and includes a serial number used to track fixed content shipping containers.
	UPC-A bar codes are used to identify all retail products in the US and Canada. You must purchase your product number assignment from gs1.org or other sources to make this kind of bar code.

INSTALLATION

Please use our RetailCD.exe program to install this package and font set in Windows. The fonts will be installed into your Windows fonts folder automatically. Our bar code utility program, Retail.exe, documentation, and matching files will also be installed. The bar code fonts will appear in all Windows font menus. You can select any one of the bar code just as you normally select any other font in your application. For help in building bar codes, run the Retail.exe utility program. To access the utility program or the documentation, click on Start, Programs, Retail Bar Codes, and select the program or documentation item you want to view.

RETAIL UTILITY PROGRAM



The Retail.exe utility program was automatically added to your system when you ran our install program. You can access the Retail.exe utility from the Desktop icon, or by clicking on: Start, Programs, Retail Bar Codes, Utility program. This program lets you enter your data and build any retail bar code. You can print a sample page of the resulting bar code, transfer that bar code to another Windows program, print sheets of labels, or export the bar code as a bmp, gif, jpg, or png file.

The Retail Bar Code Solution builds every type of retail bar code you could ever need. Just select the type of bar code you want to make, type your data, then click *Make* to build your bar code. Use this utility to make bar codes and then to copy and paste them into other Windows programs. Or print sheets of labels, selecting exactly what size labels you have. Bar codes can be copied into Word for label printing, or into a desktop publishing program to display a bar code on your packaging. These bar code fonts are compatible with virtually any Windows program. For programs like Excel and Access see the documentation section on Visual Basic macros for details on using them directly inside one of these programs.

- (1) **Type of bar code.** Chose the type of bar code you want to make here. UPC-A or EAN-13 for retail products. ISBN / Bookland for books. ITF-14 for multiple items in a box. And finally SCC-14 or SSCC-18 for shipping container codes.
- (2) **Bar code options.** You can control basic bar code options here. Switch between human readable and standard bar codes. Human readable bar codes print the data in the bar code below the bar code patterns. Add or remove the closing box ends from ITF-14 bar codes. Or control the optional Price Code in an ISBN bar code.

- (3) **Change bar code height.** Modify the height of a bar code without changing the width. Select one of the font sizes shown here.
- (4) **Change bar code width.** Scale the bar code width up or down by increasing or decreasing the point size.
- (5) **Help.** Launch the Windows Help application. Help will give you detailed answers to the most common questions about how to use this program.
- (6) **Program information.** Display program, contact information, and the version number of this package.
- (7) **Show a Tip of the Day.** Display the next program Tip of the Day. From this display you can page forwards or backwards through the list of all available tips.
- (8) **Enter bar code data here.** Type in the data you want to convert to a bar code here. Retail bar codes only use numbers. Once you enter your data, click the *Make* button to build your bar code.
- (9) **Make a bar code.** Type in the data you want to convert to a bar code here. Retail bar codes only use numbers. Once you enter your data, click the *Make* button to build your bar code. No other function will work until after you click the *Make* button.
- (10) **Display data for font.** This is the string of characters to use with our bar code fonts. Note you can not just type in most bar code fonts. Your bar code data is converted into a data string and this data string must be used with our bar code fonts.
- (11) **This is your bar code.** This is a sample of what your bar code will look like. Note that Windows screen resolution is fairly low, so the bar code displayed on screen may not exactly match what will be printed.
- (12) **Copy to clip board.** Copy your completed bar code to the Windows clip board. (*The Windows clip board is invisible- don't worry about where it is or how it works.*) You can then paste this bar code directly into any Windows program by pressing Ctrl-V or using the Edit, Paste menu in your program.
- (13) **Print sheets of labels.** Let's you print a sheet of labels. Click to open a window and select your label size. Our program then opens a label template in your word processor. Paste the bar code into any label and add text or graphics.
- (14) **Print a sample bar code sheet.** Prints a sample page showing the bar code you just built in a number of different sizes.
- (15) **Export.** Export a copy of your bar code in bmp, gif, jpg, or png formats. You can choose from four different resolutions. You can email this bar code or use it in your packaging.
- (16) **Check for updates.** Go on the web to check for program updates

USER ACCESSIBLE FILES

In Windows Vista and Windows 7, plus all future versions of Windows, users are not allowed to access any files stored in the Programs folder. The Visual Basic macros, label templates, and sample Excel spread sheet are placed into a folder named "Retail" in each user's Documents area. You can find these files as follows:

Windows XP: My Documents\Retail

Vista: Libraries\Documents\Retail

Windows 7: Libraries\Documents\Retail

EAN BAR CODES

Typeface Name	EAN a	EAN b	EAN c	EAN d	EAN e
EAN-13	 2 923457 890247	 2 923457 890247	 2 923457 890247	 2 923457 890247	 2 923457 890247
Typeface Name	EAN a plain	EAN b plain	EAN c plain	EAN d plain	EAN e plain
EAN-13					

EAN-13 bar codes are used to identify retail products in all countries except the US and Canada. You must purchase a number assignment from gs1.org or other sources to make this kind of bar code. EAN bar codes always have a Start character, data, a guard bar, more data, a checksum, and a Stop character. The 5 digit supplemental bar code is a bit more complicated. . If you just want to print a few bar codes in your existing Windows programs, use our Retail.exe utility program. (*Go back to the Retail.exe section.*) If you want to use Access, Excel, or Word to print bar codes, see the section on Visual Basic macro functions.

EAN bar codes represent the numbers 0-9 with three different bar code patterns. Thus, the bar code patterns for a 0 in Tall A are different from the bar code patterns for a 0 in Tall B. Tall C bar code patterns are the inverse images of Tall A patterns. Bar code data is always encoded using two or three different bar code patterns (*Tall A & C for EAN 8, and Tall A, B, & C for EAN 13*). Table 1 identifies which ASCII characters represent the individual numbers 0-9 for each bar code pattern. Our bar code fonts include representations of the Tall A, B, and C bar code patterns, Start codes with leading numbers plus Start codes without a number, and Short A & B patterns for the 5 digit supplemental code.

Table 1


Start		Tall A		Tall B		Tall C		Short A		Short B	
Number	Character Position	Number	Character Position	Number	Character Position	Number	Character Position	Number	Character Position	Number	Character Position
0	!	0	A	0	Q	0	0	0	a	0	q
1	"	1	B	1	R	1	1	1	b	1	r
2	#	2	C	2	S	2	2	2	c	2	s
3	\$	3	D	3	T	3	3	3	d	3	t
4	%	4	E	4	U	4	4	4	e	4	u
5	&	5	F	5	V	5	5	5	f	5	v
6	'	6	G	6	W	6	6	6	g	6	w
7	(7	H	7	X	7	7	7	h	7	x
8)	8	I	8	Y	8	8	8	i	8	y
9	*	9	J	9	Z	9	9	9	j	9	z
Start	[Stop]	Guard		Start 5	/	Space 5	.		

EAN 13 bar codes always begin with a special numbered *Start code*. This is just a Start character with a number from 0-9 printed to the left of the start pattern. The number represents the left-most digit of the bar code. (*Numbers 0-9 are represented by the sequential ASCII characters !-**) **Note** that the Start Character does not actually encode any data. In Excel or Access use our Visual Basic function, EAN13().

The Start code is followed by 6 data characters using a *mixture* of the Tall A & B characters (*numbers 0-9 are represented by ASCII characters A-J and Q-Z respectively*), a Guard Bar |, 5 more data characters and one checksum character using the Tall C characters (*numbers 0-9 are represented by ASCII characters 0-9*), and a *Stop code*].

An EAN 13 bar code has room for only 12 digits of bar coding data, but actually represents a 13-digit number. The 13th digit (*the left-most digit*) is encoded by using a pattern of different bar codes (*Tall A & B*) to represent five of the significant digits (*left side digits 11-7*) of the bar code. The right side of the bar code (*digits 6-1*) is encoded using Tall C. The 12th digit is always encoded in Tall A.

To make an EAN 13 bar code of the 12 digits of data “292345789024”, we calculate a checksum digit (7) and we encode:

Position:	13	12	11	10	9	8	7	6	5	4	3	2	1
Data:	2	9	2	3	4	5	7	8	9	0	2	4	7
Start/Left/Right	2	9	23457					890247					
Map characters	#	J	CTUFX					890247					
Plus Start, Stop, etc	#	J	CTUFX					890247]				
Complete string	#JCTUFX 890247]												
Bar Code													

Note that the left digit group “23457” does not use just the Tall A or the Tall B characters to represent the number “23457”. In this example, the 2 uses Tall A, 3 & 4 use Tall B, 5 uses Tall A, and 7 uses Tall B. The reason for this is that the 13th digit (a “2”), is encoded in the pattern of switching from the Tall A font to the Tall B font! To select the encoding for these 5 digits use Table 2. Look up the value of the 13th digit of your bar code in this table. That row in the table will tell you whether to use Tall A, or Tall B to encode each of the data digits 11 – 7. In our example, the 13th digit is a 2. From the table we see that Digit 11 should be encoded using Tall A, Digit 10 is Tall B, Digit 9 is Tall B, digit 8 is Tall A, and Digit 7 is Tall B. So Digit 11, a “2”, uses Tall A where a “2” is represented by the letter “C”. Digit 10, a “3”, uses Tall B where a “3” is represented by the letter “T”. (*These values come from Table 1.*)

Table 2

Digit 13	Digit 11	Digit 10	Digit 9	Digit 8	Digit 7
0	A	A	A	A	A
1	A	B	A	B	B
2	A	B	B	A	B
3	A	B	B	B	A
4	B	A	A	B	B
5	B	B	A	A	B
6	B	B	B	A	A
7	B	A	B	A	B
8	B	A	B	B	A
9	B	B	A	B	A



EAN 13
are calculated
Assign the
digit an “odd”
then alternate
assignments
of the digits.

checksums
as follows.
right-most
value, and
even/odd
with the rest
Add all the

odd position digits and multiply that by 3. Add to that the sum of all the even position digits. The checksum is the smallest number that can be added to that sum to make it an even multiple of 10.

Data: 2 9 2 3 4 5 7 8 9 0 2 4
Position: E O E O E O E O E O E O
Odd sum: 9 + 3 + 5 + 8 + 0 + 4 = 29, 29 * 3 = 87
Even sum: 2 + 2 + 4 + 7 + 9 + 2 = 26
Odd plus even: 87 + 26 = 113
Checksum: 7 (113 + 7 = 120, which is an even multiple of 10)

ISBN / BOOKLAND

Typeface Name	EAN a	EAN b	EAN c	EAN d	EAN e
EAN-13					

ISBN or Bookland bar codes are printed on the back of books. They have two parts: an ISBN number and an optional book price. Currently, Bookland bar codes always start with the number “978”, followed by the left-most 9 digits of the ISBN number. (*The right-most digit of the ISBN number is a checksum and it is not used.*) As of January 1, 2007, all ISBN numbers will be 13 digits long and always start with 978, or when those numbers run out, 979. The main portion of a Bookland bar code is simply an EAN 13 bar code. This includes the “978” or “979” and the 10 digit ISBN number. The optional pricing information is carried in a 5 digit supplemental code, positioned to the right of the EAN 13 bar code. In Excel or Access use our Visual Basic function, Bookland() or BooklandnoSup().

As stated above, a Bookland bar code is actually composed of two separate bar codes. The first bar code is an EAN 13 bar code that represents the book’s ISBN number. The second piece of the Bookland bar code is an optional 5 digit supplemental code. The first digit of the supplemental code indicates the currency the price is stated in. (*The number 5, for example, indicates the book is priced in US dollars.*) The next 4 digits show the price of the book. So a book that costs \$12.95 would have a 5

digit supplemental code of “51295”. Books priced at \$99.99 or higher are encoded as “59999”. Finally, if no price is stated for the book, a code of “90000” is used.

To make a Bookland bar code, you first have to convert your ISBN number and your pricing information into the proper format. Take your ISBN number and drop the right-most (*least significant*) digit. Place the numbers “978” before your ISBN number. So if your ISBN number was 1-56830-289-4, you drop the “4”, add “978”, and you get a number “978156830289”. (*If you are using our Retail.exe utility, you just enter the ISBN number. The utility figures out the rest.*) You encode this 12 digit number as a standard EAN 13 bar code. (*The checksum is the 13th digit.*) See the previous section on EAN 13 encoding for details.



Next, translate your book price into a 5 digit supplemental code. If your book sells for \$9.95 in US dollars, your supplemental number is 50995 (*5 for US dollars, and 0995 for the price*). Finally, you need to calculate a parity pattern for this supplemental number. Assign the right-most digit an “odd” value, and then alternate even/odd assignments with the rest of the digits. Add all the odd position digits and multiply that sum by 3. Add all the even position digits and multiply that by 9. Add those two sums together. The parity pattern is the number in the units position (*or the remainder when you divide by 10*).

```
Data:           5 0 9 9 5
Position:       O E O E O
Odd sum:        5 + 9 + 5 = 19, 19 * 3 = 57
Even sum:       0 + 9 = 9, 9 * 9 = 81
Odd plus even:  57 + 81 = 138
Parity Pattern: 8
```

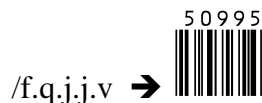
The five digit supplemental code uses the Short A and Short B bar code patterns (*from Table 2*). You select the bar code pattern (*Short A or B*) for each digit of the supplemental code by looking up the parity pattern you just calculated in Table 3. For the 5 digit supplemental code “50995” we calculated a parity pattern of 8. Looking up the “8” in Table 3, we get a value of “ABAAB”. That means the first (*left-most*) digit (5) is encoded in Short A, the “0” in Short B, the first “9” in Short A, the next “9” in Short A, and the final “5” in Short B.

Table 3

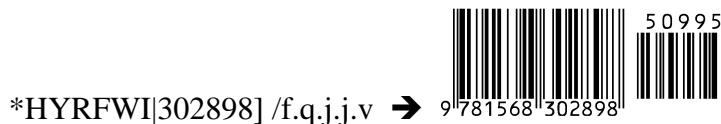
Number	Pattern
0	B B A A A
1	B A B A A
2	B A A B A
3	B A A A B
4	A B B A A
5	A A B B A
6	A A A B B
7	A B A B A
8	A B A A B
9	A A B A B



The last step is to put these bar code patterns together into a full five digit supplemental code. The supplemental code must be separated from the EAN13 bar code by a single space. It must have its own Start character, which is the “/”, and every digit must be separated from its neighbor by a character delineator- the period, “.”. For our example data (50995, parity pattern 8) this gives us the string:



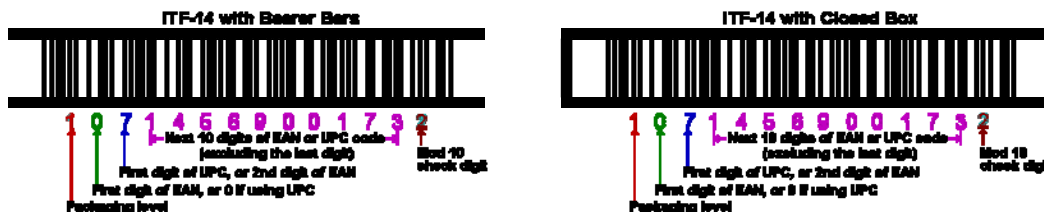
To put this all together, add the two strings, placing a space between them. The result is:



ITF-14 BAR CODES

Typeface Name	ITF14 b	ITF14 c	ITF14 d	ITF14 e	ITF14 f
ITF-14	 0 0 7 1	 0 0 7 1	 0 0 7 1	 0 0 7 1	 0 0 7 1

ITF-14 bar codes are 14 digits long. The first digit is your packaging level. It indicates the package variant. For example, suppose you package the same product in two different cases and one case contains 10 items while the other case contains 12 items. You then assign 1 for the first case and 2 for the second. There is no standard mandating that how many items should be placed in the case. A 0 is typically used if there are no package levels. If you are encoding a UPC number your next digit is a 0. If you are encoding an EAN number this digit is the leading digit from your EAN code. The next digit is the first number of your UPC code or the 2nd number of your EAN code. Then grab the next 10 digits of either your UPC or EAN code, excluding the last digit. The final digit is a mod 10 checksum, calculated as shown below. You can build a quick ITF-14 bar code from your UPC-A bar code by adding two leading zeros. The checksum will remain the same in this case (assuming you start with two 0’s). This is **not** the case in the example below!



Checksums are calculated as follows. Take the 13 digit ITF number and assign the right-most digit an “odd” value. Next, alternate even/odd assignments for the rest of the digits. Add all the odd position digits and multiply that sum by 3. Add to that first sum the sum of all the even position digits. The checksum is the smallest number that can be added to that sum to make it an even multiple of 10.

Data: 1 0 7 1 4 5 6 9 0 0 1 7 3
 Position: O E O E O E O E O E O E O
 Odd sum: 1 + 7 + 4 + 6 + 0 + 1 + 3 = 22, 22 * 3 = **66**
 Even sum: 0 + 1 + 5 + 9 + 0 + 7 = **22**
 Sum odd+even: 66 + 22 = **88**
 Checksum: **2** (88 + 2 = 90, which is an even multiple of 10)

Bar Code 2 of 5 is a high density, self-checking bar code format. It achieves this high density printing by assigning a single bar code pattern to every *pair* of digits from 00 through 99. Thus, bar code 2/5 interleaved can not represent a single digit number like a 4. It can only represent a number with two digits, like 04. There are 102 bar code patterns in the font: a Start code, a Stop code, and 100 bar codes for the pairs of digits 00 through 99. (*You must use an even number of digits in your bar code. If the code contains an odd number of digits you must add a leading 0.*)

Table 4
Bar Code 2/5 Interleaved ASCII Position vs Number Pair

ASCII Pos Char		Number Pair	ASCII Pos Char		Number Pair	ASCII Pos Char		Number Pair	ASCII Pos Char		Number Pair
33	!	00	59	;	26	85	U	52	111	o	78
34	"	01	60	<	27	86	V	53	112	p	79
35	#	02	61	=	28	87	W	54	113	q	80
36	\$	03	62	>	29	88	X	55	114	r	81
37	%	04	63	?	30	89	Y	56	115	s	82
38	&	05	64	@	31	90	Z	57	116	t	83
39	'	06	65	A	32	91	[58	117	u	84
40	(07	66	B	33	92	\	59	118	v	85
41)	08	67	C	34	93]	60	119	w	86
42	*	09	68	D	35	94	^	61	120	x	87
43	+	10	69	E	36	95	_	62	121	y	88
44	,	11	70	F	37	96	`	63	122	z	89
45	-	12	71	G	38	97	a	64			
46	.	13	72	H	39	98	b	65	123	{	Start
47	/	14	73	I	40	99	c	66	125	}	Stop
48	0	15	74	J	41	100	d	67			
49	1	16	75	K	42	101	e	68	161 / 194	ı or Â	90
50	2	17	76	L	43	102	f	69	162 / 195	ç or Ã	91
51	3	18	77	M	44	103	g	70	163 / 196	£ or Ä	92
52	4	19	78	N	45	104	h	71	164 / 197	œ or Å	93
53	5	20	79	O	46	105	i	72	165 / 198	¥ or Æ	94
54	6	21	80	P	47	106	j	73	166 / 199	ı or Ç	95
55	7	22	81	Q	48	107	k	74	167 / 200	§ or È	96
56	8	23	82	R	49	108	l	75	168 / 201	¨ or É	97
57	9	24	83	S	50	109	m	76	169 / 202	© or Ê	98
58	:	25	84	T	51	110	n	77	170 / 203	ª or Ë	99

Encoding the data is done in the same way you do it for standard bar code 2 of 5 interleaved numbers:

UPC-A number: 714569001735
 ITF-14 number: 1 + 0 + 71456900173 + 2 → 10714569001732
 Data encoded: +hNf!2A
 Start and Stop pattern: { } |~{ }~|
 Data string: {+hNf!2A} |~{+hNf!2A}~|

SCC-14

Typeface Name	Code 128C a	Code 128C b	Code 128C c	Code 128C d	Code 128C e	Code 128C f
SCC-14						



(01)	0	0	712345	12345	9
Application Identifier	Packaging Type	Use a leading 0 for UPC codes or	First 6 digits Of UPC code	5 digit item identification number from UPC	Mod 10 Check digit
Do not enter In our utility Or macros	0 = box/carton 1 = pallet 2 = larger	1 st digit of EAN code			We calculate, Do not add

The Shipping Container Code, (SCC-14) is used on fixed content shipping containers. Note that this bar code only contains numbers. It has 14 variable digits, and an additional first two that must always be 01. The last digit is a checksum. The human readable portion shown above is formatted for people to read. The actual bar code does not include and spaces or parenthesis. Assume your UPC code is **712345123459**. To create an SCC-14 bar code from your UPC code:

- 1) Start with the packaging type, enter a 0 for a box (or a 1 or a 2 for other carton sizes)
- 2) Enter a second 0, followed by the first digit of your UPC code
- 3) Enter the next 5 digits of your UPC code
- 4) Enter the next 5 digits of your UPC code- leaving out the last digit of your UPC
- 5) Our program will calculate the checksum of 9
- 6) So your data would be: 0 + 0 + 7 + 12345 + 12345 + 9 = **00712345123459**

SSCC-18

Typeface Name	Code 128C a	Code 128C b	Code 128C c	Code 128C d	Code 128C e	Code 128C f
SSCC-18						

The Serial Shipping Container Code, (SSCC-18) bar code is used as an identifier on shipments to track items. Note that this bar code only contains numbers. It has 18 digits, but the first two must always be 00 and the last digit is a checksum. The human readable portion shown above is formatted for people to read. The actual bar code does not include and spaces or parenthesis. Assume your UPC code is **712345678904**. To create an SSCC-18 bar code from your UPC code:



(00)	0	0	712345	123456789	7
Application Identifier	Extension digit	Use a leading 0 plus 1 st digit of UPC code	6 digit UPC company ID number	9 digit serial number	Mod 10 Check digit
	Set to 0	or first digits of EAN code	or digits 2 – 7 of EAN code	not to be used again for 12 months	

- 1) Start with a leading 0 extension digit
- 2) Enter a second 0, followed by the first digit of your UPC code
- 3) Enter the next 5 digits of your UPC code, leaving out the last 6 digits of your UPC
- 4) Enter a 9 digit serial number
- 5) Our program will calculate the checksum of 7
- 6) Your data would be: $0 + 0 + 7 + 12345 + 123456789 + 7 = \mathbf{007123451234567897}$

UPC-A

Typeface Name	UPCA a	UPCA b	UPCA c	UPCA d	UPCA e
UPC-A					
	7 14569 00173 5	7 14569 00173 5	7 14569 00173 5	7 14569 00173 5	7 14569 00173 5

Typeface Name	UPC Plain a	UPC Plain b	UPC Plain c	UPC Plain d	UPC Plain e
UPC-A					

UPC-A bar codes are used to identify all retail products in the US and Canada. You must purchase a number assignment from gs1.org (expensive), mybarcodestore.com, or other sources to make this kind

of bar code. If you just want to print a few bar codes in your existing Windows programs, use our Retail.exe utility program. (*Go back to the Retail.exe section.*) If you want to use Access, Excel, or Word to print bar codes, see the section on our Visual Basic functions. UPC-A bar codes always have a Start character, 5 data characters, a guard bar, 5 more data characters, and a checksum / Stop character.



UPC-A bar codes represent the numbers 0-9 with two different bar code patterns. Half of the bar code goes to the left of a “guard” bar, and the other half goes to the right of it. The bar code patterns for a 0 on the left side of the bar code are the mirror image of the bar code patterns for a 0 on the right side of the same bar code. In addition, the height of the bar code Start and Stop characters is different from the height of the rest of the data in the bar code. Table 5 identifies which ASCII characters represent the individual numbers 0-9 for each bar code pattern. Our bar code fonts also include representations of Short Odd and Even patterns for the 2 and 5 digit supplemental codes.

Table 5

|← **UPC-A Codes** →| |← 2 & 5 supplemental →|
digit

Start		Left / Odd		Right / Even		Stop		Short Odd		Short Even	
Number	Character Position	Number	Character Position	Number	Character Position	Number	Character Position	Number	Character Position	Number	Character Position
0	!	0	A	0	0	0	Q	0	a	0	q
1	"	1	B	1	1	1	R	1	b	1	r
2	#	2	C	2	2	2	S	2	c	2	s
3	\$	3	D	3	3	3	T	3	d	3	t
4	%	4	E	4	4	4	U	4	e	4	u
5	&	5	F	5	5	5	V	5	f	5	v
6	'	6	G	6	6	6	W	6	g	6	w
7	(7	H	7	7	7	X	7	h	7	x
8)	8	I	8	8	8	Y	8	i	8	y
9	*	9	J	9	9	9	Z	9	j	9	z
Start	[Stop]	Guard		Start 5	/	Space 5	.		

UPC-A bar codes always begin with a special numbered *Start Code*. This is a Start character combined with a leading digit from 0-9. A human readable number is printed to the left of the Start pattern. This number represents the left-most digit of the bar code. (*Start numbers 0-9 are represented by the sequential ASCII characters ! through *.*)

The Start Code is followed by 5 data characters using the Left/Odd characters (*numbers 0-9 are represented by ASCII characters A through J*), a Guard Bar |, 5 more data characters using the Right/Even characters (*numbers 0-9 are represented by ASCII characters 0 through 9*), and a checksum character using the Stop characters (*numbers 0-9 are represented by ASCII characters Q through Z*).

To make a UPC-A bar code of the 11 digits of data “71456900173”, we calculate a checksum digit (5) and we encode:

Position: 12 11 10 9 8 7 6 5 4 3 2 1
 Data: 7 1 4 5 6 9 0 0 1 7 3 5

Start/Left/Right/Checksum 7 14569 00173 5
Add guard bar **7 14569 | 00173 5**
 Map characters (BEFGJ | 00173 V
 Complete string (BEFGJ|00173V



Bar Code

UPC-A checksums are calculated as follows. Take the 11 digit UPC number and assign the right-most digit an “odd” value. Next alternate even/odd assignments for the rest of the digits. Add all the odd position digits and multiply that sum by 3. Add to that first sum the sum of all the even position digits. The checksum is the smallest number that can be added to that sum to make it an even multiple of 10.

Data: 7 1 4 5 6 9 0 0 1 7 3
 Position: O E O E O E O E O E O
 Odd sum: 7 + 4 + 6 + 0 + 1 + 3 = 21, 21 * 3 = 63
 Even sum: 1 + 5 + 9 + 0 + 7 = 22
Sum odd plus even: 63 + 22 = 85
 Checksum: 5 (85 + 5 = 90, which is an even multiple of 10)

CONTROLLING BAR CODE HEIGHT AND WIDTH

Each bar code font set includes five to six different aspect ratios of each bar code: 0.25 (*A*), 0.5 (*B*), 0.75 (*C*), 1.0 (*D*), 1.25 (*E*) & 1.5 (*F*). When you are building a bar code, start with the C or D version from our bar code fonts. Change the width of the bar code by varying the point size of the bar code font. Once you have the bar code width set to what you want, you can vary the bar code height, without changing the bar code length, by changing the font to one of the A, B, C, D, or E versions. The B version will print a bar code of exactly the same length as the D, but at .75 times the bar height. The E version will print a bar code of exactly the same length as the D, but at 1.25 times the bar height.

PRINTING SHEETS OF LABELS

This bar code font set adds the ability to print bar codes to other, existing Windows programs. It does not print bar code labels all by itself. It uses label templates to lay out label sheets in your existing word processor. A template tells your word processor how to arrange information to fit specific sheets of labels. Once you have a template you can use any word processor to build your own labels with bar codes. The bar codes do not have to be identical and you can also include other text or graphics on each label. Our utility will let you create a bar code and then automatically open up the label template of your choice in your word processor.

Just click on *Labels*, select a label size, and click *Print*. Your word processor will open and display the label template. Put the cursor in the first label and paste your bar code there. Adjust the size of the bar code to fit the label by changing the font name and point size.

This package includes label templates in RTF format (*compatible with virtually all word processors*) for most popular label styles. You can open any of these templates in your word processor to lay out a label. To locate these files using your word processor click on Open and then select the “Libraries\Documents” folder. Look for a new subfolder with the name: \Retail.

Template	Label Size	Labels/Sheet
ef167.rtf	1.75 x 0.50	80
ef570.rtf	1.75 x 1.25	32
ef060.rtf	2.625 x 1.00	30
ef161.rtf	4.00 x 1.00	20
ef162.rtf	4.00 x 1.33	14
ef163.rtf	4.00 x 2.00	10
ef197.rtf	4.00 x 1.50	12
ef164.rtf	4.00 x 3.33	6

Our Retail.exe utility also lets you use your own custom label template. See your word processor manual for details on how to create a template, or download one from your label supplier. Once you have your own custom template, save the template file (*in rich text format*) in the Libraries\Documents\Retail folder with the file name “custom.rtf”. Our Retail.exe utility will directly access that template to let you easily print labels.

EMBEDDING BAR CODE FONTS IN PDF FILES

The fonts in this bar code set are **not** embeddable. When you embed a font in a PDF document, you are actually distributing a copy of that font with every single PDF file you generate. Your PDF generator glues a copy of the bar code font to the end of each PDF file it builds. This font is then installed on every computer that views the PDF document.

Embeddable versions of these fonts are available, at additional cost. Pricing for this add-on font set is based on how many computers the embeddable fonts will be installed on and how many people will view the PDF files. When you purchase an embeddable add-on font set you receive a new version of these fonts, and the new fonts will embed in PDF files. You must contact Elfring Fonts to order an embeddable font set.

VISUAL BASIC MACROS FOR ACCESS, EXCEL, AND WORD

This package contains Visual Basic macros that let you automatically build Retail bar code strings in Excel, Access, and indirectly in Word. These macros, and sample Excel spread sheets in both Excel 2003 and Excel 2010 formats, are located in a folder named Retail under your Libraries\Documents folder. (**Libraries\Documents\Retail**)

BarRetail.xls Excel 2003 sample spread sheet
 BarRetail.xlsm Excel 2010 sample spread sheet
 EFretail.bas Visual Basic source code

Table 6

Function	Details
BarEAN13(Number)	Converts the input 12 digit number into a complete EAN 13 bar code. The function adds the Start code, appends the data and guard bar, calculates and adds the checksum, and puts the Stop code at the end. This result must be formatted with one of the following typefaces: EAN a, EAN b, EAN c, EAN d, EAN e. Non-numeric data is ignored. If less than 12 digits are supplied, all are replaced with zeros. Any digits past the twelfth place are ignored.
Bookland(ISBN,Supplem)	Converts the input ISBN number into a complete Bookland bar code. The function converts the ISBN number into EAN 13 format and adds the 5 digit supplemental code to the end. This result must be formatted with one of the following typefaces: EAN a, EAN b, EAN c, EAN d, EAN e. Non-numeric data is ignored. If the ISBN number is less than 9 digits it will be replaced with zeros. Any digits past the ninth place are ignored. The supplemental code must be 5 digits long.
BooklandnoSup(ISBN)	Converts the input ISBN number into a Bookland bar code. The function converts the ISBN number into EAN 13 format. This result must be formatted with one of the following typefaces: EAN a, EAN b, EAN c, EAN d, EAN e. Non-numeric data is ignored.
BarITF14(Text)	Converts the input 13 or 14 digit number into ITF-14 format. The function adds the bounding box, Start code, breaks the data up into number pairs and converts them to single characters, adds the checksum digit and then puts the Stop code at the end. Non-numeric data is ignored. You must enter 13 or 14 digits. This result must be formatted with one of the following typefaces: ITF14 b, ITF14 c, ITF14 d, ITF14 e, or ITF14 f.
BarSCC14(Text)	Converts the input text data into a complete SCC-14 bar code. This function adds the Start code, the FNC1, the AI of 01, throws away all non-numeric data, converts the numeric data into number pairs, calculates and adds the mod 10 and mod 103 checksums and then the Stop character. Your data must be exactly 11 digits long.
BarSCC14checkdigit(Text)	Returns just the mod 10 SCC-14 check digit
BarSSCC18(Text)	Converts the input text data into a complete SSCC-18 bar code. This function adds the Start code, the FNC1, the AI of 00, throws away all non-numeric data, converts the numeric data into number pairs, calculates and adds the mod 10 and mod 103 checksums and then the Stop character. Your data must be exactly 15 digits long.
BarSSCC18checkdigit(Text)	Returns just the mod 10 SSCC-18 check digit
BarUPCa(Text)	Converts the input 11 or 12 digit number into a complete UPC-A bar code compatible with our UPC-A fonts. The function adds the Start code, translates and appends the data plus guard bar, calculates and adds the checksum, and puts the Stop code at the end. Non-numeric data is ignored, as are numbers greater than 12 digits in length. This result must be formatted with one of the following typefaces: UPCA a, UPCA b, UPCA c, UPCA d, UPCA e.
BarUPCcheck(Text)	Take an 11 digit UPC-A number and calculate the checksum for it. Add the checksum to the end of the 11 digit number and return the resulting 12 digit UPC-A bar code number. (<i>Not a valid bar code string, just the actual bar code number including the checksum.</i>)

Warning!

Unlike our utility program, *Retail.exe*, these Visual Basic functions do **very** limited error checking. You must make sure that the data you send to the function is correct! If you send bad data, your bar

codes may be unreadable or they may not encode the data you think they have. Please use the *Retail.exe* utility program to verify that your data is correct before using these Visual Basic functions to mass produce bar codes.

Using Visual Basic Functions in Excel

Open the spread sheet where you want to add bar code functions (*or create a new spread sheet*). Click on Tools, Macros, then Visual Basic Editor. In the Visual Basic Editor tool, click on File, Import File, and select the drive and user data folder. (Libraries\Documents\Retail or My Documents\Retail) The Visual Basic file, **EFretail.bas**, should appear there. Select this file and open it. This will add a new function module, EFRetail, to your spread sheet. This module is automatically saved along with your spread sheet and adds the macro functions shown in Table 6.

These bar code functions can be used in any formula or cell to build working bar codes. For example, if cell H9 is defined as a text cell (*Format, Cell, Number, Text*) and cell I9 has the formula, =EAN13(H9), then any text entered in cell H9 will be converted into a bar code string in cell I9. Note that you also need to select the proper typeface for that bar code type, using Format, Cell, Font. See that bar code description for applicable font names.

Using Visual Basic Functions in Access

Open the database where you want to add EAN bar code functions (*or create a new database*). Under your database Objects, click on Modules, then click on the New icon at the top of the box. This will bring up the Visual Basic Editor tool. Click on File, Import File, and select the drive and user data folder where you installed our Retail Bar Code. (Libraries\Documents\Retail or My Documents\Retail) The Visual Basic file, **EFretail.bas**, should appear there. Select this file and open it. This will add a new function module, EFRetail, to your database. This module is automatically saved along with your database and adds the macro functions shown in Table 6.

These bar code functions can be used in a report to build working bar codes. To use these functions, pass data to them from your table fields and return the result in a report. You do this by entering a formula =EAN13 ([table.field]) in the control source field of the report. Note that you need to select the proper typeface for that bar code subset. See that bar code's description for a complete listing of font names.

Using Visual Basic Functions in Word

While Word does have the ability to use Visual Basic functions in macros, you need to be a macro expert to use these functions. You can, however, combine Word and Excel to print Retail bar codes! The basic idea is to keep your data in Excel, and to have Excel format the data for bar code printing. You then do a mail merge in Word, selecting the formatted string from Excel as the data source.

Place your data in Excel in a specific column. Now apply one of our bar code functions to the data in that column, producing a second column that contains the formatted string. This second column is passed to Word in the mail merge. You must make sure to select the proper subset typeface for your mail merge field. See that bar code's description for a complete list of font names.

Glossary

Application Program: a computer program that performs useful work not related to the computer itself. Examples are word processors, spreadsheets, accounting systems, and engineering programs.

ASCII: American Standard Code for Information Interchange – a standard code for representing characters as numbers that is used on most microcomputers, computer terminals, and printers.

Aspect ratio: the ratio of the bar code height to bar code width.

Bookland: a universal bar code for books that converts a book's ISBN number into an EAN 13 bar code and adds a 5 digit supplemental code for the book price.

Character string: a sequence of characters stored in a computer and treated as a single data item.

Checksum: a number that accompanies data transferred from one place to another and helps to ensure that the data was transferred correctly.

Clipboard: a holding area to which information can be copied in order to transfer it from one application to another.

CPI: characters per inch: the number of characters a fixed width font will print per inch.

DPI: dots per inch: the number of pixels or printer dots per linear inch.

EAN: European Article Numbering- the retail bar code used everywhere except North America.

Font: a complete collection of characters, in a consistent style and size. This includes upper and lower case letters, numerals, punctuation, ligatures, and reference marks.

ITF-14: an implementation of the Global Trade Information Number (GTIN)

Modulo: a mathematical operation that gives the remainder when one number is divided by another.

Number Assignment: Retail products must have unique ID numbers. To keep these identifying numbers unique, they are all assigned by a single agency, GS1. You can not make up your own numbers.

Parity pattern: a clever way of encoding extra data, like a checksum, in the individual characters of a bar code.

Pixels: one of the individual dots that make up a graphical image.

Scalable: able to be used on a large or small scale without major changes.

Scalable font: a font that can be used to print characters of any size. Many newer laser printers include scalable fonts.

SCC-14: Shipping Container Code- a bar code for fixed content shipping containers.

SSCC-18: Serial Shipping Container Code- a bar code for fixed content shipping containers that also includes tracking information.

Symbology: a method of representing information by printed characters.

OpenType Font: an outline typeface that can be scaled or sized to practically any size.

Scanner: a device that enables a computer to read printed or handwritten page.

Start Code: the leading character of a bar code that identifies what kind of bar code it is.

Stop Code: the last character of a bar code that tells the bar code reader when it has reached the end of the code.

Utility: a program that assists in the operation of a computer but does not do the main work for which the computer was bought.

UPC: Uniform Product Code- the bar code used on all retail products in North America

Visual Basic Function: a small program written in Visual Basic and used in products like Excel, Access, or Word to help create bar codes.

Copyright 1992-2011 by Elfring Fonts, Inc.
2020 Dean St, Unit N
St Charles, IL 60174
Phone: 1-630-377-3520 Fax: 1-630-377-6402
www.barcodingfonts.com
www.mybarcodestore.com