

# Barcodes not recognized – What can you do?

A support for QS-Beleg, QS-Barcode, and bcTester

---

## About linear barcodes in general

A linear barcode is a sequence of bars (dark lines) and spaces between the bars (white lines). A linear barcode always starts and ends with a bar, a bar is always followed by a space, and a space is always followed by a bar. The width of the bars and spaces are usually different, and the order of the bars and spaces along with their width carry the information encoded by the barcode.

There are many different barcode symbologies (codings) each of which uses different orders and widths. If a barcode symbology distinguishes two different widths for the bars and two different widths for the spaces (thin and narrow), the symbology is binary, otherwise the symbology is non-binary. Many barcode symbologies use checksums or checksum characters.

In addition to the usual linear barcode described above, the new 2D (two dimensional) barcodes are increasingly used. Their elements are arranged in a plain.

**Usual linear barcode (1D)**  
content: 6 characters



**2D barcode**  
content: 57 characters



The system of 2D barcodes enables to compress larger contents on smaller spaces.

The most common 2D Barcodes are the PDF 417 and the Data Matrix Code. These barcodes can be created in many different sizes and can contain more than 2, 000 characters, which can also be encoded. Advertising mail also often contains data in 2D barcodes .

# Barcodes not recognized – What can you do?

A support for QS-Beleg, QS-Barcode, and bcTester

---

## Barcodes not recognized - What can you do?

Occasionally, a barcode is not immediately recognized by "QS-Barcode". This does not necessarily mean that the barcode cannot be recognized by „QS-Barcode“. In most cases, the reasons for unsuccessful recognition are errors which can easily be corrected.

For an analysis of recognition problems, use our Freeware **bcTester**, which you can find on the internet (<http://www.bctester.de/en/>).

You should have your non-recognized barcodes as an image file. If you have problems loading the file into the bcTester program, you may have to choose another format for your image.

In general the reasons for a non-successful barcode recognition can be divided into three main groups:

- **1. "Inappropriate" scanner settings**
- **2. Errors in printing the barcode**
- **3. Inappropriate parameters for recognition**

In the following chapters you will find hints to help you avoid the most frequently occurring errors.

If this is not successful, please contact QualitySoft.

**We will analyze your barcodes for free!**

**Just send your images and a short description of your problem to  
[support@qualitysoft.de](mailto:support@qualitysoft.de)**

For further information about our products, please visit our website

<http://www.qualitysoft.de>

# Barcodes not recognized – What can you do?

A support for QS-Beleg, QS-Barcode, and bcTester

## 1. Cause of Error: "Inappropriate" scanner settings

Often, barcode errors are already visible in the image. The following section outlines typical errors based on some examples. Here is an ideal situation:



This code only has 2 kinds of thicknesses, this means:

### Bar thickness:

- all thin bars have the same width
- all thick bars have the same width
- the thick bars are three times as wide as the thin ones

### Gap thickness:

- all thin gaps have the same width
- all thick gaps have the same width
- the thick gaps are three times as wide as the thin ones

### Ratio bars/gaps:

- the thin bars have the same width as the thin gaps
- the thick bars have the same width as the thick gaps



This code has 4 kinds of thicknesses, this means:

### Bar thickness:

- Four different kinds of bar thicknesses are possible. They can be twice, three or four times as wide as the thinnest bar.

### Gap thickness:

- Four different kinds of gap thickness are possible. They can be twice, three or four times as wide as the thinnest bar.

### Ratio bars/gaps:

The different kinds of bar thickness match the thicknesses of the gaps.

In comparison to the printed original, scanned images are always faulty, e.g. gaps and dots on the edge, step curves, etc. In a good image these errors are very rare, the bars and gaps should comply with the description above.

### Good Scan:



Code 39: 123456

# Barcodes not recognized – What can you do?

A support for QS-Beleg, QS-Barcode, and bcTester

If the resolution during scanning is too low, fine bars cannot be separated. Two or more bars become one and the gaps disappear.

**Scan resolution too low (50 dpi):**



If the scan is too dark, the bars enlarge which makes thin gaps disappear. The barcode cannot be recognized. If the barcode cannot be scanned in a brighter way (otherwise other bright elements may disappear from the image) the barcode must be printed much wider.

**Too dark (-75)**



If the scan is too bright, bars which are necessary for the recognition can be missing.

**Too much brightness: (+85):**



Here you see some examples of barcodes, which are not readable because of wrong scanner resolutions (outlines, dithering, high jpg compression):



# Barcodes not recognized – What can you do?

A support for QS-Beleg, QS-Barcode, and bcTester

## 2. Cause of Error: Errors in printing the barcodes

In practice, errors in printing barcodes will occur which makes recognition impossible. Obvious mistakes include unequal widths of bars and gaps which should be the same. In printing on labels, the barcodes are often printed too close to the edge. This way the light margins on the right and left side of the barcode or even whole bars of the barcode are missing. The same can happen with some print programs, if the field for the barcode is too small. In addition, you should never write or stamp on a barcode, this can also make a recognition impossible.

If the barcode appears ok on the scanned image but cannot be read, this may be because the start and stop characters were forgotten during printing. With many barcodes you can easily count whether all the necessary bars are contained:

We will explain the general layout of a barcode using Code 39 as an example. In this code, 9 elements (5 bars and 4 gaps) form one character, three of them are thick.

**Characters of Code 39** (only digits are shown, letters A-Z and the special characters \$/+% can be encoded with Code 39 as well).



The \* (star) character is used as a start and stop character. This means that the number 1234 is correctly encoded as \*1234\* and then consists of  $6 \times 5 = 30$  bars. For clarification the following picture shows every single sign, the start and stop characters are red.



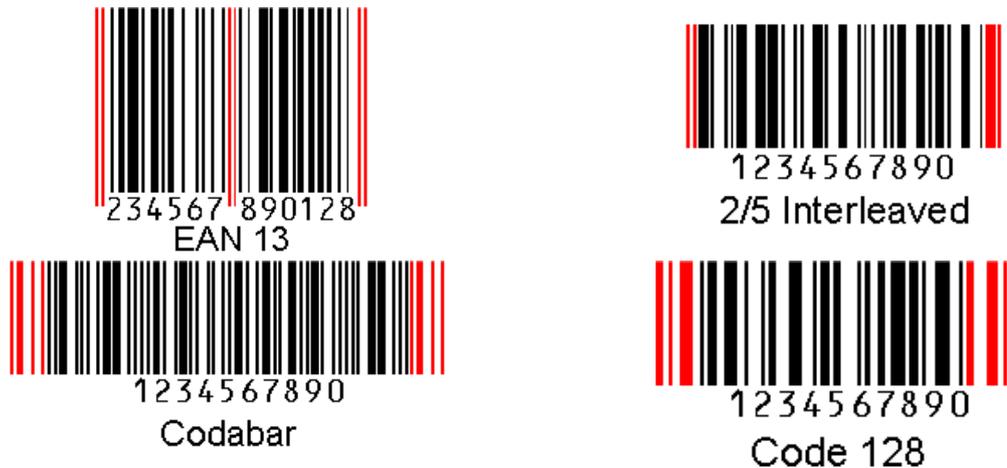
Without the gaps the code looks like this:



# Barcodes not recognized – What can you do?

A support for QS-Beleg, QS-Barcode, and bcTester

The start and stop characters of some major barcode types:



The start / stop characters must be used! Every type has specific characters which help to identify the barcode type. Calculating the number of bars a barcode contains depends on the type and the characters of the barcode. The formula is not always as easy as in Code 39, because there are certain types which contain compressions, checksums, control and escape characters.

## 3. Cause of Error: Inappropriate parameters for recognition

It may be necessary to change the standard parameters before recognition to find the optimal settings for the image you wish to recognize. This chapter explains which changes in the parameters may be helpful.

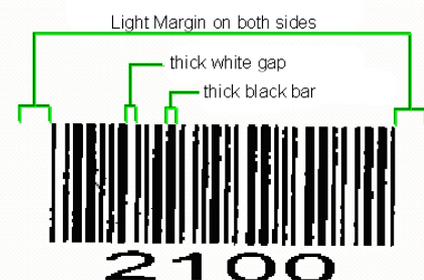
### Proceeding

Perform first tests with the standard settings of bcTester, only then begin to change single parameters in the menu "Barcode Settings". The following section explains some of these parameters in detail.

### Light Margin

On both sides of a barcode a white area is expected as a light margin. This white area must be wider than any white gap contained in the barcode and wider than any black bar contained in it (q.v. picture on the right).

*Is the present light margin wider than the widest gap contained in the barcode?*

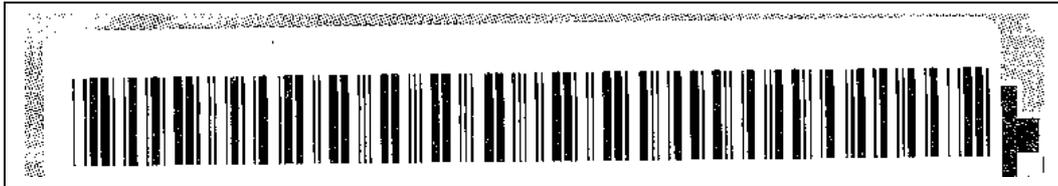


# Barcodes not recognized – What can you do?

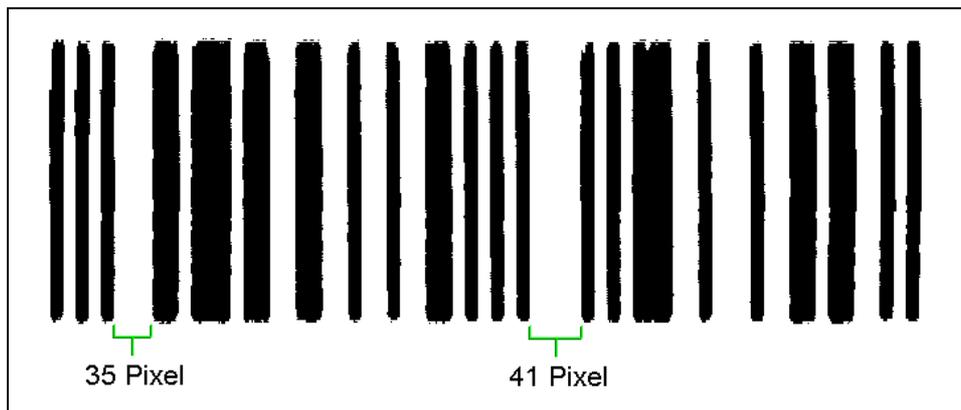
A support for QS-Beleg, QS-Barcode, and bcTester

In bcTester, you find the light margin in the dialog “Barcode Settings”, <Extended Parameters>. Here you can choose its size in pixels. The standard setting of 18 pixels refers to the common 200dpi images. If you scanned with a high resolution, you should try a higher value.

The following example shows a barcode label where the barcode has been printed too close to the edge: the light margin behind the barcode is too small. As a result, the “t” printed under the label is recognized as an element of the barcode and a correct recognition is impossible. In this case, reducing the light margin would not be helpful either, because then the white gaps contained in the barcode would also be recognized as light margins.



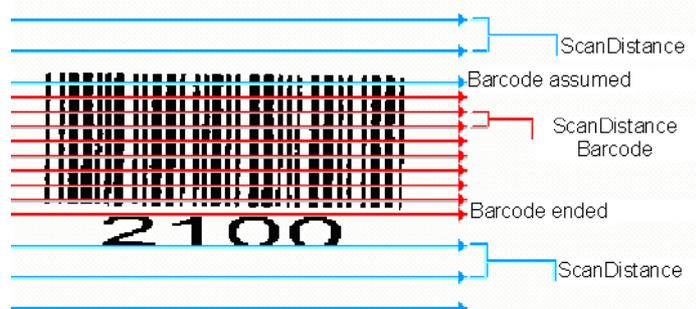
The following example was scanned with 600dpi. The high scanner resolution leads to white gaps contained in the barcode whose width is far beyond the standard value 18 pixels. If you increase the light margin value to 45 pixels, the barcode can be read without any problems.



## ScanDistance and ScanDistanceBarcode

The algorithm for Barcode Search searches the image row by row. In most cases it is not necessary to analyze every single row, which saves a lot of time.

Usually a “ScanDistance” of 15 rows is set. If a barcode is assumed, the algorithms changes to “ScanDistanceBarcode” (standard value: 3 rows) to collect more information inside the barcode.



# Barcodes not recognized – What can you do?

A support for QS-Beleg, QS-Barcode, and bcTester

As a result, the “ScanDistanceBarcode” should always be smaller than or as large as the “ScanDistance”. Both parameters can be set in the dialog “Barcode Settings”, <Extended Parameters>.

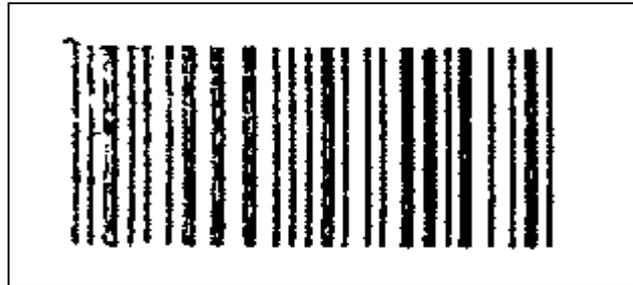
To achieve optimal recognition results, choose 1 for both parameters. In this way *every* row of the image is read individually. Of course, this slows down the recognition speed. Thus these parameter settings are useful for testing purposes: if the recognition is successful you can enhance the values step by step.

## Number of Barcodes: BC\_ONE\_BREAK (ReadMulti)

In special cases, especially if barcodes are printed or scanned in a poor quality, you can try recognition using the special parameter “BC\_ONE\_BREAK”.

As always, the image is read row by row. As soon as a barcode is recognized in one row, the recognition is stopped immediately and the barcode is reported as the result.

The pros and cons of this setting are described in the following table.



### Pros

The setting „BC\_ONE\_BREAK“ allows a fast recognition because the reading is stopped very early

The setting „BC\_ONE\_BREAK“ allows the recognition of highly damaged barcodes even if only one row of the barcode is readable

### Cons

Only one barcode can be found

It is possible that fonts or gray scales are wrongly interpreted as a barcode which is followed by a *wrong* result! If you choose other settings, this error is eliminated by analyzing several rows to check the result.

In several cases of use, it is better to achieve no results than errors. In this cases you definitely should not use the setting “BC\_ONE\_BREAK”.

The specification of an exact length, orientation and type reduces the error rate.

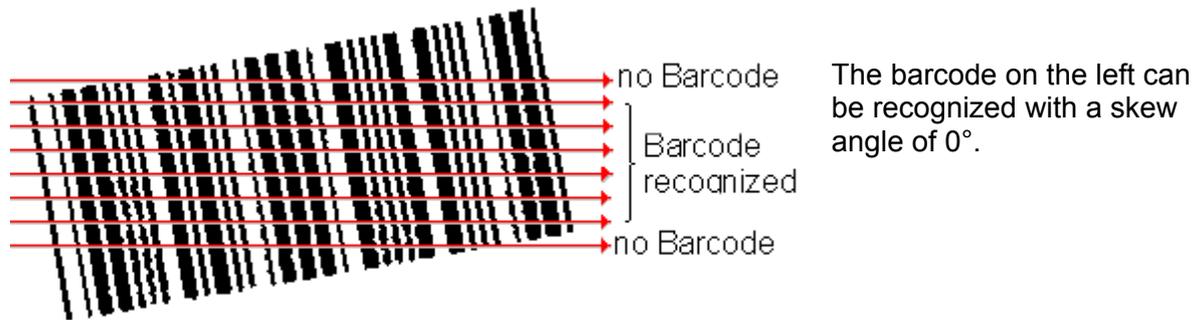
The barcode example above is of very poor quality; the first bars in particular are highly damaged. With this quality the used barcode should contain a checksum, so that erroneous results can easily be noticed!

# Barcodes not recognized – What can you do?

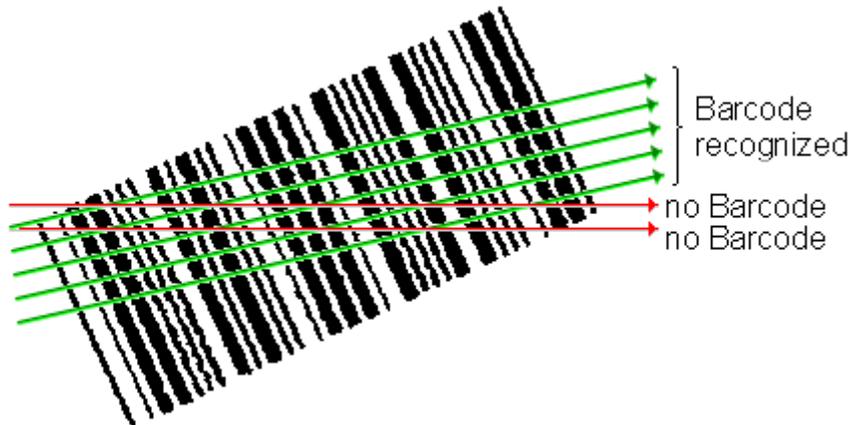
A support for QS-Beleg, QS-Barcode, and bcTester

## Rotations

With standard settings, barcodes are searched with  $0^\circ / 180^\circ$  and  $90^\circ / 270^\circ$  rotation. Small variations up to  $8^\circ$  - as they emerge e.g. during scanning - are tolerated. If barcodes occur with a much higher rotation, the parameter "maximum skew angle" should be set to  $21^\circ$ ,  $34^\circ$  or  $46^\circ$ . If you choose  $46^\circ$ , all directions are searched.



The barcode on the right cannot be recognized with a "skew angle" of  $0^\circ$  (**red arrows**), but with a "skew angle" of  $21^\circ$ , because then the image is also searched diagonally (**green arrows**).



QS QualitySoft GmbH  
Zum Fürstenmoor 11  
21079 Hamburg, Germany

Tel: +49 40 790 100 40

[info@qualitysoft.de](mailto:info@qualitysoft.de)

[www.qualitysoft.de](http://www.qualitysoft.de)

© 2004 QS QualitySoft GmbH