

# ReportLab Barcode Test Suite - full output

Generated on Tue Sep 27 20:38:10 2016  
Repository information for this build:

## About this document

### History and Status

This is the test suite and documentation for the ReportLab open source barcode API, being re-released as part of the forthcoming ReportLab 2.0 release.

Several years ago Ty Sarna contributed a barcode module to the ReportLab community. Several of the codes were used by him in his work and to the best of our knowledge this was correct. These were written as flowable objects and were available in PDFs, but not in our graphics framework. However, we had no knowledge of barcodes ourselves and did not advertise or extend the package.

We "wrapped" the barcodes to be usable within our graphics framework; they are now available as Drawing objects which can be rendered to EPS files or bitmaps. For the last 2 years this has been available in our Diagra and Report Markup Language products. However, we did not charge separately and use was on an "as is" basis.

A major licensee of our technology has kindly agreed to part-fund proper productisation of this code on an open source basis in Q1 2006. This has involved addition of EAN codes as well as a proper testing program. Henceforth we intend to publicise the code more widely, gather feedback, accept contributions of code and treat it as "supported".

This involved making available both downloads and testing resources. This PDF document is the output of the current test suite. It contains codes you can scan (if you use a nice sharp laser printer!), and will be extended over coming weeks to include usage examples and notes on each barcode and how widely tested they are. This is being done through documentation strings in the barcode objects themselves so should always be up to date.

### Usage examples

To be completed

### The codes

Below we show a scannable code from each barcode, with and without human-readable text.

These are magnified about 2x from the natural size done by the original author to aid inspection.

This will be expanded to include several test cases per code, and to add explanations of checksums. Be aware that (a) if you enter numeric codes which are too short they may be prefixed for you (e.g. "123" for an 8-digit code becomes "00000123"), and that the scanned results and readable text will generally include extra checksums at the end.

**Scale = 1.0**

**Code: Codabar**



Barcode should say: A012345B

**Code: Code11**



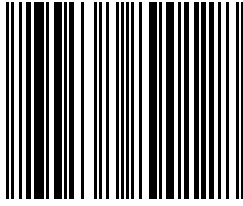
Barcode should say: 01234545634563

**Code: Code128**



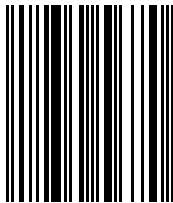
Barcode should say: AB-12345678

## Code: EAN13



Barcode should say: 123456789012

## Code: EAN8



Barcode should say: 1234567

## Code: Extended39



Barcode should say: A012345B}

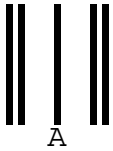
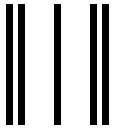
## Code: Extended93



L@@K! Code 93 ;- )4K

Barcode should say: L@@K! Code 93 ;-)

## Code: FIM



A

Barcode should say: A

## Code: I2of5



012348

Barcode should say: 1234

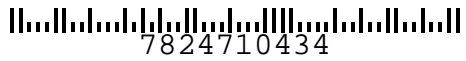
## Code: MSI



12344

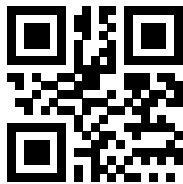
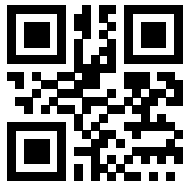
Barcode should say: 1234

## Code: POSTNET



Barcode should say: 78247-1043

## Code: QR



Barcode should say: Hello World

## Code: Standard39



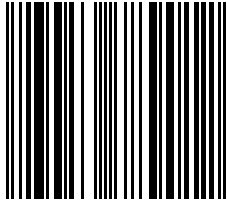
Barcode should say: A012345B%R

## Code: Standard93



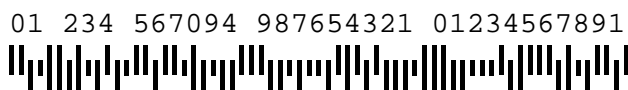
Barcode should say: CODE 93

## Code: UPCA



Barcode should say: 12345678901

## Code: USPS\_4State



Barcode should say: 01 234 567094 987654321 01234567891