

Stephan Kerbei

Wie der Computer lesen lernte

Das 1x1 der Barcodes

Eine Grundvoraussetzung des Lesens ist es, Buchstaben und Ziffern an ihrer Form eindeutig zu erkennen. Bei gedrucktem Text ist das zumeist noch recht einfach, aber bei Handschriften wird es manchmal schon zu einem Problem. Ist das jetzt eine 1 oder eine 7 – oder gar eine 4? Wenn das für uns Menschen schon schwierig ist, um wie viel komplizierter ist es dann für einen „dummen“ Computer? Zwar gibt es Texterkennungsprogramme, die Buchstaben anhand ihrer Form identifizieren; aber sie sind noch zu aufwendig und fehleranfällig, um damit z.B. an einer Supermarktkasse die Preise einzulesen.

Eine einfachere Möglichkeit, einen Computer lesen zu lassen, wurde bereits in den 70er Jahren entwickelt: Der *Strichcode* oder engl. *Barcode* (Abbildung 1). Ein Lesestift oder ein Scanner wird über unterschiedlich breite schwarze und weiße Streifen geführt und registriert den Wechsel zwischen hell und dunkel wie ein Bitmuster. Ist nun bekannt, welches Muster welchem Buchstaben oder welcher Ziffer entspricht, kann man das Muster problemlos in den Klartext rückübersetzen.

Mittlerweile sind eine ganze Reihe solcher Strichcodes für verschiedene Zwecke entwickelt worden. Vom rein numerischen über den alphanumerischen bis hin zu einem Code, der den kompletten ASCII-Zeichensatz abdeckt. Es gibt Codes mit fester Länge und Codes, die beliebig lang sein können. Bei einigen sind Prüfziffern enthalten, bei anderen nicht. Für jeden Bedarf ist etwas dabei.

Es reicht aber nicht aus, einen Strichcode zu definieren. Um ihn einzusetzen, braucht man auch Lesegeräte, die den Code in Klartext rückübersetzen können. Natürlich kann man dem Anwender nicht zumuten, viele verschiedene Lesegeräte für viele verschiedene Codes einzusetzen. Daher wurde es notwendig, sich auf bestimmte Strichcodes für den jeweiligen Einsatzbereich (z.B. Produktauszeichnung

oder Paketaufkleber) zu einigen. Das führte dazu, dass eine Reihe von Codes sehr schnell wieder verschwunden ist, während sich andere zu Standard-Codes entwickelt haben.

Artikelnummern mit Strichcodes kodieren

Eines der Einsatzgebiete, in dem sich der Strichcode schnell einen festen Platz erobern konnte, ist das Erkennen von Produkten. Sei es beim Wareneingang oder, wo wir es täglich sehen können, beim Einscannen der Artikel-



Abbildung 1:
Der EAN13-Code der BasicPro.

nummer an der Kasse eines Supermarktes. Für diesen Zweck hat sich europaweit seit 1977 der *EAN-Code* (*European Article Numbering*) und in den USA und Kanada bereits seit 1973 der

ÜBER blick

Thema

Strichcodes sind aus unserem Alltag nicht mehr wegzudenken. Auf Lebensmittelpackungen, Büchern und vielen anderen Dingen des täglichen Bedarfs begegnen uns diese „Zebrastreifen“. Wie aber sind die gängigsten Strichcodes überhaupt aufgebaut? Welche Informationen können sie enthalten? Und wie erzeugt man sie mit VB-Bordmitteln? Das sind Fragen, die dieser Grundlagenartikel beantwortet.

VB-Versionen: VB5, VB6

Zielgruppe

Alle VB-Entwickler, die an Grundlagen interessiert sind; Entwickler, die Software mit möglichst wenigen externen Steuerelementen/Komponenten ausliefern möchten.

UPC (*Universal Product Code*) durchgesetzt. Beide definieren nicht nur, wie der Strichcode aussieht, sondern sorgen auch dafür, dass die Artikelnummern zentral vergeben werden. Damit ist sichergestellt, dass es jede Nummer nur einmal gibt und eine Kasse somit zweifelsfrei zwischen Gummibärchen und Taschentüchern unterscheiden kann. Auch viele Länder außerhalb Europas und Nordamerikas haben sich dem EAN-System angeschlossen, sodass wir mittlerweile schon von einem weltweit eindeutigen System der Artikelnummerierung sprechen können.

Sowohl bei EAN als auch bei UPC gibt es einen Standard-Code und mehrere leicht davon

abweichende Versionen für besondere Fälle – falls zum Beispiel für den Standard-Code nicht genug Platz auf einer Verpackung vorhanden ist oder sehr kurze Zahlen dargestellt werden müssen.

Die folgenden Abschnitte stellen die unterschiedlichen Strichcode-Definitionen vor und gehen auf deren Besonderheiten und Einsatzgebiete ein – denn Strichcode ist nicht gleich Strichcode, auch wenn sie für uns Menschen irgendwie alle wie „Zebrastrreifen“ aussehen.

Der EAN13

Der für EAN hauptsächlich verwendete Code ist der EAN13. Hierbei handelt es sich um einen Code, der nur numerische Werte darstellen kann, mit einer festen Länge von – der Name lässt es schon vermuten – 13 Ziffern. Die EAN13-Nummer für die BasicPro zum Beispiel lautet 4393060019805.

Aufbau

Die ersten drei Stellen bezeichnen das Land, in dem der Hersteller eines Produktes seinen Sitz hat. Der Nummernkreis 400 bis 440 zum Beispiel steht für Deutschland.

Der Rest der Ziffernfolge wird von jedem angeschlossenen Land selbst verwaltet. Zuständig für Deutschland ist die CCG (Centrale für Coorganisation GmbH) mit Sitz in Köln. Sie vergibt an den Hersteller eine 4-stellige Herstellernummer, und dieser wiederum gibt

seinem Artikel eine 5-stellige Artikelnummer.

Als Letztes kommt noch eine Prüfziffer hinzu, sodass der EAN13 wie folgt aufgebaut ist:

Land (3) + Hersteller (4) + Artikelnummer (5) + Prüfziffer (1)

Nummern, die mit einer 2 beginnen, sind in dem System der Ländernummern absichtlich nicht vorhanden. Diese Nummern dürfen von jedermann für eigene Nummerierungen frei genutzt werden. Sie sind dann allerdings nicht eindeutig. Eine (sicher unvollständige) Liste der Ländernummern finden Sie in *Tabelle 1*.

Die Prüfziffer

Die dreizehnte Ziffer des EAN13 ist die Prüfziffer, die aus den ersten zwölf Ziffern berechnet werden kann. Dazu müssen diese zwölf Ziffern *von hinten nach vorne* betrachtet werden:

Zunächst wird die Summe der Ziffern an den *ungeraden* Positionen, also den Positionen 1, 3, 5 usw. (von *hinten an* gezählt!) gebildet, danach die Summe der Ziffern an den *geraden* Positionen 2, 4, 6 usw. Die *ungerade* Summe wird nun mit 3 multipliziert und zu der *geraden* Summe addiert. Die Zahl, die nun noch addiert werden müsste, damit das Ergebnis mit einer 0 endet, ist die Prüfziffer.

An die folgende Visual Basic-Funktion zur Berechnung dieser Prüfziffer wird ein 12-stelliger String übergeben. Das Ergebnis (die Prüfziffer) ist ein 1-stelliger String, der sofort an diese Ziffernfolge angehängt werden kann.

Nummer	Land	Nummer	Land
00-09	USA und Kanada	599	Ungarn
20-29	interne Nummerierungen	600-601	Südafrika
30-37	Frankreich	609	Mauritius
380	Bulgarien	611	Marokko
383	Slowenien	613	Algerien
385	Kroatien	619	Tunesien
387	Bosnien-Herzegowina	64	Finnland
400-440	Deutschland	690-691	China
460-469	Russland	70	Norwegen
471	Taiwan	729	Israel
474	Estland	73	Schweden
475	Lettland	746	Dominikanische Republik
477	Litauen	750	Mexiko
479	Sri Lanka	759	Venezuela
480	Philippinen	76	Schweiz
482	Ukraine	770	Kolumbien
489	Hong Kong	773	Uruguay
45+49	Japan	775	Peru
50	Großbritannien	789	Brasilien
520	Griechenland	80-83	Italien
529	Zypern	84	Spanien
531	Mazedonien	850	Kuba
535	Malta	858	Slowakische Republik
539	Irland	859	Tschechische Republik
54	Belgien und Luxemburg	860	Jugoslawien
560	Portugal	869	Türkei
569	Island	87	Niederlande
57	Dänemark	90-91	Österreich
590	Polen	93	Australien
594	Rumänien	94	Neuseeland

Tabelle 1: Ländernummern bilden die ersten drei Stellen von EAN13-Strichcodes. Eine 2-stellige Ländernummer bedeutet, dass die dritte Stelle beliebig ergänzt werden kann; z.B. kann Australien (93) Nummern von 930 bis 939 verwenden.

Strichcodes in diesem Artikel

Die Strichcode-Abbildungen zu diesem Artikel können nicht von einem Lesegerät gelesen werden. Das ursprünglich lesbare Strichmuster ist, bis es letztendlich in diesem Heft gelandet ist, durch mehrere Text- und Grafik-Programme gelaufen und dabei so häufig gestaucht und gestreckt worden, dass die Strichbreite nicht mehr einheitlich ist – sie ist aber die Voraussetzung für die Lesbarkeit.

Wenn Sie jedoch das Beispielpogramm auf der Heft-CD benutzen und damit Strichcodes drucken, dann haben diese selbstverständlich die nötige Qualität und werden korrekt von einem Barcode-Scanner gelesen.

Kasten 1: Achtung beim Versuch, die Strichcodes in diesem Heft zu scannen

```
Public Function _
EAN13Prüfziffer(ByVal _
strZiffernfolge As String) As String
Dim intSum_ungerade As Integer
Dim intSum_gerade As Integer
Dim intErgebnis As Integer
Dim i%, j%

intSum_gerade = 0
intSum_ungerade = 0
j% = 0
For i% = Len(strZiffernfolge) To 1 Step -1
' von hinten nach vorne
j% = j% + 1
If j% Mod 2 = 1 Then
intSum_ungerade = _
intSum_ungerade + _
Val(Mid$(strZiffernfolge, i%, 1))
Else
intSum_gerade = intSum_gerade + _
Val(Mid$(strZiffernfolge, i%, 1))
End If
Next i%
intErgebnis = intSum_ungerade * 3 _
+ intSum_gerade
intErgebnis = 10 - intErgebnis Mod 10
If intErgebnis = 10 Then intErgebnis = 0

EAN13Prüfziffer = Right$(Str$(intErgebnis), 1)
End Function
```

Als Beispiel soll hier einmal die EAN-Nummer der BasicPro dienen. Die zwölf Ziffern, die zum Berechnen der Prüfziffer herangezogen werden, lauten 439306001980.

Die ungerade Summe errechnet sich also:
 $0 + 9 + 0 + 6 + 3 + 3 = 21$.

Die gerade Summe: $8 + 1 + 0 + 0 + 9 + 4 = 22$.

Ungerade Summe * 3 = 63.

$63 + 22 = 85$.

Um zur nächsten glatt durch 10 teilbaren Zahl, nämlich der 90, zu gelangen, müssen noch 5 addiert werden, und genau diese 5 ist die

Prüfziffer für die EAN-Nummer unserer Zeitschrift, wie sie *Abbildung 1* zeigt.

Die Prüfziffer wird deshalb von hinten nach vorne berechnet, weil so die Berechnung der Prüfziffern für eine ganze Reihe anderer Standard-Strichcodes genau identisch durchgeführt werden kann, unabhängig davon, ob die Stellenzahl gerade oder ungerade ist.

Die Kodierung der Ziffern im EAN13

Um eine Zahl in einen Strichcode umzuwandeln, muss es für jede Ziffer eine ganz bestimmte Folge von Strichen geben. Im EAN13 gibt es sogar drei verschiedene Kombinationen von Strichen pro Ziffer. Das heißt, zu jeder Strichkombination gibt es genau eine Ziffer, aber jede Ziffer kann auf drei verschiedene Arten dargestellt werden. Warum das so ist, werden wir später noch sehen. Die drei möglichen Formate werden als *Format A*, *Format B* und *Format C* bezeichnet, manchmal aber auch als *Format Links A*, *Format Links B* und *Format Rechts*.

Der Strichcode einer einzelnen Ziffer besteht aus zwei schwarzen Balken und zwei Leeräumen, die jeweils eins bis drei Segmente breit sind und gemeinsam sieben Segmente bilden. Das klingt recht kompliziert. Glücklicherweise gibt es aber noch eine andere Definition, die von einem Programmierer leichter verstanden wird, wahrscheinlich, weil sie nahezu unmittelbar in einen Programmcode übernommen werden kann:

Die einzelne Ziffer besteht aus genau sieben Strichen, die entweder schwarz oder weiß sind. Da es keine Lücken zwischen den einzelnen Strichen gibt, wirken zwei oder sogar drei schwarze Striche nebeneinander wie ein dickerer Balken. Mehrere weiße Striche unmittelbar nebeneinander erscheinen dem Auge als größerer Leerraum. Welcher Strich weiß und welcher schwarz ist, ist für jede Ziffer in jedem der drei Formate genau festgelegt. Die Reihenfolge lässt sich in Form eines Bitmusters darstellen. Für die Ziffer 4 in Format A wäre das zum Beispiel die Folge 0100011, wobei eine 0 für einen weißen, eine 1 für einen schwarzen Strich steht. Das Muster, das sich aus dieser Bitfolge ergibt, ist in *Abbildung 2* dargestellt.

Um die einzelnen Ziffern klar voneinander und von den Start- und Endezeichen abgrenzen zu können, müssen alle Zeichen in der linken

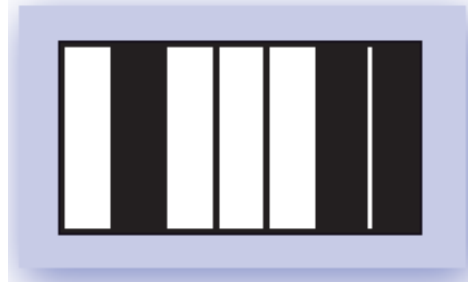


Abbildung 2: Schema der Ziffer 4 im Format A. Jeder schwarze Strich steht für eine 1, jeder weiße für eine 0 im Bitmuster dieses Formats.

Hälfte des Codes mit einem weißen Strich beginnen und mit einem schwarzen enden. In der rechten Hälfte muss es genau umgekehrt sein. Es sind also verschiedene Formate für die Ziffern der linken und der rechten Seite des Codes erforderlich.

Eine komplette Aufstellung der Bitmuster für den EAN-Code zeigt die *Tabelle 2*. Das Format C ist das Komplement des Formats A: Wo im Format A eine 0 steht, befindet sich im Format C eine 1. Das Format B ist das Spiegelbild des Formats C.

Sonderzeichen

Darüber hinaus gibt es noch zwei Sonderzeichen: Das Zeichen für Anfang und Ende des Strichcodes, 101, und die Kombination 01010 in der Mitte des Strichcodes. Anfangs- und Endezeichen sind erforderlich, damit das Lesegerät, egal, ob es den Code von links nach rechts oder von rechts nach links liest, immer zunächst auf das Muster „schwarz – weiß – schwarz“ stößt. Nur anhand dieses Musters kann es erkennen, wie breit ein einzelner Strich tatsächlich ist.

Das Mittelzeichen wird benötigt, um ein Umschalten zwischen linken und rechten Zeichen zu ermöglichen. Die Doppelstriche dieser Sonderzeichen werden häufig etwas länger gedruckt als die Zifferncodes (siehe *Abbildung 1*). Das ist allerdings keine technische Notwendigkeit, sondern dient einzig und allein der optischen Verschönerung des Druckbildes.

Die Strichfolgen für die ersten, linken sechs Ziffern des Codes werden immer in einem der Formate A oder B dargestellt, die für die letzten, rechten sechs Ziffern immer im Format C.

Ziffer	Format A	Format B	Format C
0	0001101	0100111	1110010
1	0011001	0110011	1100110
2	0010011	0011011	1101100
3	0111101	0100001	1000010
4	0100011	0011101	1011100
5	0110001	0111001	1001110
6	0101111	0000101	1010000
7	0111011	0010001	1000100
8	0110111	0001001	1001000
9	0001011	0010111	1110100

Tabelle 2: Bitmuster für die Ziffern in den verschiedenen Formaten des EAN-Codes.

Ziffer	Kombination	Ziffer	Kombination
0	AAAAAA	5	ABBAAB
1	AABABB	6	ABBBA
2	AABBAB	7	ABABAB
3	AABBBA	8	ABABBA
4	ABAABB	9	ABBABA

Tabelle 3: Kombination von Formaten für die sechs Ziffern an Position 2 bis 7, abhängig von der ersten Stelle der EAN13-Nummer.

Implizite Kodierung der ersten Ziffer

Halt! Da stimmt doch etwas nicht! Sechs Ziffern vorne, sechs hinten – das sind zusammen zwölf. Die scheinbar fehlende Ziffer, es ist die erste der 13-stelligen Zahl, wird nicht durch einen Strichcode, sondern durch eine Kombination von Formaten dargestellt.

Der einzige Grund für diese Vorgehensweise liegt darin, dass dadurch der EAN13-Code kompatibel zum UPC-A-Code bleibt, der gleich noch vorgestellt wird. Abhängig von der ersten Ziffer der EAN-Nummer wird festgelegt, in welchem Format (A oder B) die folgenden sechs Ziffern dargestellt werden. *Tabelle 3* zeigt die in Frage kommenden Kombinationen.

Bei der EAN-Nummer der BasicPro (*Abbildung 1*) ergibt sich der Strichcode also nach folgendem Muster (Die Schreibweise „4A“ soll dabei bedeuten: Strichcode der Ziffer 4 im Format A):

Anfang – 3A – 9B – 3A – 0A – 6B – 0B – Mitte – 0C – 1C – 9C – 8C – 0C – 5C – Ende

Die erste Ziffer der EAN-Nummer (4) ist darin über die Formatkombination „ABAABB“ links von der Mitte kodiert.

Der UPC-A-Code

Der 12-stellige UPC-A-Code und der 13-stellige EAN13 unterscheiden sich nur unwesentlich voneinander. Das Verfahren zur Berechnung der Prüfziffer ist identisch. Auch die Methode zur Ermittlung des Bitmusters für die Strichdarstellung ist gleich. Es ist nur notwendig, vor die zwölf Stellen der UPC-Nummer eine Null zu hängen, um die gleichen Druckroutinen wie beim EAN13 nutzen zu können.

Bei UPC-Nummern, die mit 0, 6 oder 7 beginnen, handelt es sich um Artikelnummern, bei denen die Stellen 1-6 den Hersteller bezeichnen und die Stellen 7-11 eine vom Hersteller vergebene Nummer. UPCs, die mit einer 3 beginnen, dienen zum Kennzeichnen von Medikamenten.



Abbildung 3: Der EAN8-Code.

Die Eindeutigkeit der UPC-Nummern wird sichergestellt durch das *Uniform Code Council* mit Sitz in Dayton, Ohio.

Ein interessantes Phänomen ist, dass Lesegeräte, die den EAN13-Code erkennen können, problemlos auch den UPC-A Code lesen, was umgekehrt nicht unbedingt funktioniert. Die Ursache liegt darin, dass beim EAN13-Code das Format der ersten sechs durch Strichmuster dargestellten Ziffern von der ersten Ziffer der (immer 13-stelligen) Nummer abhängt. Beim UPC-A-Code ist das immer die 0, woraus sich laut *Tabelle 2* ergibt, dass dort niemals Zeichen im Format B vorkommen.

Der EAN8

Der Name lässt es bereits vermuten: Der EAN8-Code (*Abbildung 3*) wird verwendet, um eine 8-stellige Ziffernfolge darzustellen. Er wird für kleine Produkte genutzt, bei denen der Platz nicht ausreicht, um den relativ großen EAN13 aufzudrucken. Auch der EAN8 beginnt mit der 3-stelligen Ländernummer (siehe *Tabelle 1*) und endet mit einer Prüfziffer.

Anders als beim EAN13 kann der Hersteller die Artikelnummer aber nicht selbst vergeben, sondern die Vergabe erfolgt zentral, in Deutschland zum Beispiel durch die CCG.

Die Prüfziffer wird auf die gleiche Weise berechnet wie beim EAN13. Auch die Bitmuster für Anfang, Mitte und Ende sind identisch.

Aufbau

Die ersten vier Ziffern des Strichcodes werden nach dem Format A aus der *Tabelle 2* gebildet, die letzten vier nach dem Format C.

Der EAN2

Der EAN2-Code wird häufig neben einen EAN13 gedruckt, um eine 2-stellige Zusatzinformation zu geben. Bei der BasicPro wird er genutzt, um zusätzlich zur Artikelnummer noch die Heftnummer anzuzeigen (*Abbildung 4*).

Aufbau

Das Strichmuster des EAN2 beginnt immer mit dem Anfangszeichen 1011. Zwischen den einzelnen Strichen steht die Kombination 01, ein Endezeichen gibt es nicht. Um festzulegen, welches Strichmusterformat für die beiden Ziffern benutzt wird, muss zunächst eine kleine Rechnung durchgeführt werden.

Ist die Zahl, die dargestellt werden soll, ohne Rest durch 4 teilbar, werden beide Ziffern im Strichformat A (siehe *Tabelle 2*) gedruckt. Bleibt bei der Division durch 4 ein Rest von 1, wird die erste Ziffer im Format A, die zweite im Format B dargestellt. Beim Rest 2 ergibt sich ein Formatmuster „BA“, bei 3 „BB“.



Abbildung 4: Die Artikelnummer der BasicPro als EAN13 und daneben die Heftnummer als EAN2.

Implizite Prüfziffer

Dadurch, dass abhängig vom Wert der Zahl unterschiedliche Formate für die Darstellung der einzelnen Ziffern benutzt werden, kann man Lesefehler ausschließen, ohne Platz für eine zusätzliche Prüfziffer zu vergeuden.

Im Programm lässt sich die nötige Abfolge der Formate folgendermaßen ermitteln:

```
intWert = Val(strZeichenfolge)
intRest = intWert - (intWert \ 4) * 4
Select Case intRest
Case 0
    strMuster = "AA"
Case 1
    strMuster = "AB"
Case 2
    strMuster = "BA"
Case 3
    strMuster = "BB"
End Select
```

Der EAN5

Auch der EAN5-Code wird benutzt, um zusätzliche Informationen neben einem EAN13 anzugeben (*Abbildung 5*). Er bietet fünf Stellen, um z.B. zusätzlich zur Artikelnummer Angaben zum Preis oder Gewicht zu machen.

Aufbau

Wie beim EAN2 beginnt der Strichcode mit dem Muster 1011. Die einzelnen Ziffern werden durch 01 voneinander getrennt. Ein Endezeichen gibt es auch hier nicht.

Implizite Prüfziffer

In welchem Strichformat (A oder B) die einzelnen Ziffern dargestellt werden, ist abhängig von einer Prüfziffer, die allerdings nicht gedruckt wird, um Platz zu sparen. Auch diese Prüfziffer wird, wie immer bei Prüfzifferberechnungen im EAN-Code, von hinten nach vorne berechnet. Und genau wie bei der Berechnung der Prüfziffer für den EAN13 (siehe oben) werden zunächst die Summen für



Abbildung 5: Eine EAN13-Nummer mit EAN5-Erweiterung.

Ziffer	Kombination	Ziffer	Kombination
0	BBAAA	5	BBAAB
1	BABAA	6	BBBAA
2	BAAAB	7	BABAB
3	ABBA	8	BABBA
4	AABBA	9	BBABA

Tabelle 4: Kombination von Formaten für die Ziffern der EAN5-Nummer.

die geraden und die ungeraden Stellen gebildet. Die gerade Summe wird mit 3, die ungerade mit 9 multipliziert. Die beiden Summen werden dann addiert. Die letzte Stelle des Ergebnisses ist die Prüfziffer.

Als Beispiel soll hier die Prüfziffer für die Zeichenfolge 01395 berechnet werden:

$$\begin{aligned} \text{Ungerade Summe} &= 5 + 3 + 0 = 8 \\ \text{Gerade Summe} &= 9 + 1 = 10 \\ (8 * 3) + (10 * 9) &= 24 + 90 = 114 \end{aligned}$$

Also ist die Prüfziffer hier die 4.

Anhand der so ermittelten Ziffer und der Tabelle 4 kann die Folge der Strichformate für die auszugebende Zahl festgestellt werden. Im obigen Beispiel sieht das folgendermaßen aus:

$$0A - 1B - 3B - 9A - 5A$$

Der UPC-E-Code

Was den UPC-E-Code (Abbildung 6) betrifft, lassen sich in verschiedenen Quellen Aussagen finden, die sich teilweise erheblich widersprechen. So gibt es z.B. die Ansicht, er sei vollkommen identisch mit dem EAN8.

Tatsächlich besteht die einzige Übereinstimmung zwischen EAN8 und UPC-E darin, dass sie die gleichen Strichformate zur Darstellung der Ziffern benutzen. Bei genauem Hinsehen fällt sofort auf, dass der UPC-E nur 7 Ziffern darstellt.

Ein weiterer bedeutender Unterschied zwischen EAN8 und UPC-E ist, dass es für den EAN8 eigene 8-stellige Artikelnummern gibt, wogegen die sieben Stellen des UPC-E aus der originalen 12-stelligen UPC-Nummer herausgerechnet werden. Damit diese verkürzte Nummer immer noch eindeutig ist, ist dieses Verfahren nur für ganz bestimmte UPC-Nummern zulässig.

Aufbau von UPC-Nummern

Die UPC-Nummer setzt sich zusammen aus einer 1-stelligen Kennziffer, einer 5-stelligen



Abbildung 6: UPC-E-Code der UPC-Nummer 04710001254.

Herstellernummer, einer 5-stelligen Artikelnummer und der Prüfziffer. Um daraus eine 7-stellige UPC-E-Nummer bilden zu können, werden Nullen, die sich innerhalb der 12-stelligen Zahl befinden, eliminiert. Das wird nach folgenden Regeln durchgeführt:

- UPC-E-Codes können nur gebildet werden für Nummern mit der Kennziffer 0. Es soll in absehbarer Zeit auch einen UPC-E-Code geben, der die Kennziffer 1 zulässt. Das ist aber momentan noch nicht realisiert.
- Für Herstellernummern, die mit der Ziffernfolge 000, 100 oder 200 enden, sind Artikelnummern bis maximal 00999 erlaubt. Die UPC-E-Nummer setzt sich dann zusammen aus der Kennziffer, den ersten zwei Stellen der Herstellernummer, den letzten drei Stellen der Artikelnummer und der dritten Stelle der Herstellernummer.
- Bei den übrigen Herstellernummern, die auf 00 enden, gibt es Artikelnummern bis maximal 00099. Dann setzt sich die UPC-E-Nummer zusammen aus der Kennziffer, den drei Stellen des Herstellers, dem 2-stelligen Artikel und der Ziffer 3.
- Hersteller mit der Endziffer 0 können Artikel bis zur Nummer 00009 haben. Die UPC-E-Nummer enthält dann die Kennziffer, die ersten vier Stellen der Herstellernummer, die letzte Stelle der Artikelnummer und die 4.
- Hersteller, die keine 0 am Ende ihrer Nummer haben, können nur Artikelnummern zwischen 00005 und 00009 als UPC-E verschlüsseln. Die Formel lautet dann Kennzahl plus Herstellernummer plus letzte Stelle der Artikelnummer.

Implizite Prüfziffer

Welches Strichformat für die Ziffern benutzt wird, ist abhängig von der Prüfziffer der kompletten 12-stelligen UPC-Nummer. Die möglichen Kombinationen sind in Tabelle 5 dargestellt. Als Startzeichen ist die Strichfolge 101, als Ende die 010101 hinzuzufügen.

Code25 Interleaved

Da das EAN/UPC-System die Vergabe von international eindeutigen Artikelnummern gewährleistet, hat es sich als Strichcodesystem weitgehend durchgesetzt und dabei viele andere Systeme verdrängt. Der Code25 Interleaved (Abbildung 7), auch „Interleaved 2 of 5“ genannt, konnte sich allerdings behaupten, da er verglichen mit EAN/UPC einen Vorteil hat: Er ist nicht auf eine bestimmte Anzahl von Stellen beschränkt.

Aufbau

Die Bezeichnung „2 of 5“ bedeutet, dass jede Ziffer durch fünf Elemente gebildet wird, wobei jeweils zwei breiter sind als die übrigen. Das



Abbildung 7: Code25 Interleaved-Strichcodes sind nicht auf eine bestimmte Anzahl von Ziffern beschränkt.

Muster für die einzelnen Ziffern ist in Tabelle 5 dargestellt. Ein „B“ steht für „Breit“, ein „S“ für „Schmal“.

„Interleaved“ besagt, dass jeweils zwei Ziffern gleichzeitig ausgegeben werden, gewissermaßen miteinander vermengt: die erste Ziffer des Paares als schwarze Balken, die zweite als weiße Zwischenräume. Für das Ziffern paar 34 zum Beispiel werden die Kombinationen „BBSSS“ (für die 3) und „SSBSB“ (für die 4) gemischt und ergeben so „Bs+Sw+Bs+Sw+Ss+Bw+Ss+Sw+Ss+Bw“. „Bs“ steht für „Breit schwarz“, „Sw“ für „Schmal weiß“ usw.

In der Schreibweise, die wir bereits für alle anderen Codes benutzt haben, lässt sich die 34 darstellen als 11011010010100. Dabei wird vorausgesetzt, dass ein breiter Streifen genau doppelt so breit ist wie ein schmaler. Das muss aber nicht unbedingt so sein; die Vorschriften zur Bildung des Code25 Interleaved besagen nur, dass die breiten Symbole mindestens zweimal und höchstens dreimal so breit sein müssen wie die schmalen. Die 34 ließe sich also auch als 11011101000101000 darstellen.

Alle möglichen Bitmuster für die Ziffernfolgen 00 bis 99 sind in Tabelle 7 abzulesen.

Da immer zwei Ziffern zu einem Strichmuster zusammengefasst werden, können nur Zahlen mit einer geraden Anzahl von Ziffern dargestellt werden. Bei einer ungeraden Ziffernzahl wird eine 0 vorangestellt.

Prüfziffer optional

Eine Prüfziffer ist im Code25 Interleaved nicht vorgeschrieben. In sehr vielen Anwen-

Ziffer	Kombination	Ziffer	Kombination
0	BBBAAA	5	BAABBA
1	BBABAA	6	BAAABB
2	BBAABA	7	BABABA
3	BBAAAB	8	BABAAB
4	BABBAA	9	BAABAB

Tabelle 5: Kombination von Formaten für die Ziffern des UPC-E-Codes, abhängig von der Prüfziffer der 12-stelligen UPC-Nummer.

Ziffer	Kombination	Ziffer	Kombination
0	SSBBS	5	BSBSS
1	BSSSB	6	SBBSS
2	SBSSB	7	SSSBB
3	BBSSS	8	BSSBS
4	SSBSB	9	SBSBS

Tabelle 6: Kombination von Formaten (Breit und Schmal) für die Ziffern des Code25 Interleaved.

dungen wird trotzdem eine Prüfziffer als letzte Stelle angefügt. Sie wird in diesem Fall nach dem gleichen Verfahren gebildet wie beim EAN13 (siehe oben).

Identcode und Leitcode der Deutschen Post AG

Prüfziffer

Für die Kennzeichnung von Frachtsendungen setzt die Deutsche Post AG den Code25 Interleaved ein, allerdings mit einer etwas anderen Methode zur Berechnung der Prüfziffer. Auch hier werden (von hinten nach vorne gesehen) die Summen der geraden und die der ungeraden Stellen gebildet. Die ungerade Summe wird dann mit 4, die gerade mit 9 multipliziert und beide Summen addiert. Die Prüfziffer ist dann wiederum die Zahl, die noch addiert werden muss, um auf die „volle 10“ zu kommen.

Für die Ziffernfolge 56310243031 sieht das folgendermaßen aus:

Ungerade Summe = 1 + 0 + 4 + 0 + 3 + 5 = 13
 13 * 4 = 52
 Gerade Summe = 3 + 3 + 2 + 1 + 6 = 15
 15 * 9 = 135
 52 + 135 = 187. Bis zur vollen 10, also 190, fehlen 3, und das ist die Prüfziffer.

Identcode

Der *Identcode* kennzeichnet eine Frachtsendung eindeutig. Er wird überall, wo die Sendung umgeladen wird, automatisch erfasst, so-

dass man den Weg dieser Sendung verfolgen kann und jederzeit weiß, wo sie sich befindet.

Der Identcode setzt sich folgendermaßen zusammen:

2 Stellen Abgangsfrachtzentrum + 3 Stellen Kundennummer (von der Post vergeben) + 6 Stellen Einlieferungsnummer (vom Kunden vergeben) + Prüfziffer.

Leitcode

Der *Leitcode* ist die in Strichcode umgesetzte Anschrift des Frachtempfängers. Er ermöglicht ein automatisches Sortieren und Weiterleiten der Frachtsendung.

Der Leitcode setzt sich zusammen aus 5 Stellen Postleitzahl + 3 Stellen Straßenkennzahl + 3 Stellen Hausnummer + 2 Stellen Produktcode + Prüfziffer.

Wenn Sie z.B. ein Paket an die BasicPro-Redaktion schicken, erhält es folgenden Leitcode: 24105.056.001.00 4.

24105 ist die Postleitzahl, 056 die Kennzahl der Preußenstraße und 001 die Hausnummer. Dann kommt der Produktcode 00 für ein Paket (40 für Päckchen, 60 für Infopost Schwer) und schließlich die Prüfziffer.

Code39

Anders als die bisher beschriebenen Codes kann der Code39 (*Abbildung 8*), der 1983 im American National Standards Institute (ANSI)



Abbildung 8: Der Code39 kann nicht nur Ziffern, sondern auch alle Buchstaben und einige Sonderzeichen darstellen.

definiert wurde, auch alphanumerische Werte darstellen – wenn auch noch nicht alle.

Dieser Code, auch bekannt unter den Namen USD-3 oder 3 of 9, kann neben allen Ziffern auch die Großbuchstaben von A bis Z sowie die Zeichen Plus, Minus, Punkt, Leerzeichen, Dollar, Schrägstrich und Prozent drucken.

Aufbau

Das einzelne Zeichen besteht aus neun Elementen (fünf schwarzen und vier weißen), von denen drei breiter als die übrigen sind. Dabei müssen die breiten Elemente mindestens zweimal, aber höchstens dreimal so breit sein wie die schmalen.

Tabelle 8 zeigt die Bitmuster für die darstellbaren Zeichen, wobei hier die schmalen Elemente als ein Strich, die breiten als zwei Striche nebeneinander dargestellt sind.

Zeichentrennung

Dadurch, dass beim Code39 jedes Zeichen schwarz beginnt und auch schwarz endet, sind zwei Zeichen, die sich nebeneinander befinden, optisch nicht voneinander zu trennen. Um das zu verhindern, muss zwischen zwei Zeichen jeweils ein Leerraum eingefügt werden. Wie breit diese Lücke sein muss, ist nicht definiert; Tests haben jedoch ergeben, dass ein schmaler Strich, also eine zusätzliche 0 im Bitmuster, ein optimales Ergebnis beim Lesen bringt.

Optionale Prüfziffer

Eine Prüfzifferrechnung ist nicht zwingend vorgeschrieben. Ist sie dennoch erwünscht, wird sie normalerweise nach *Modulo43* durchgeführt. Das bedeutet, die Werte der einzelnen Zeichen werden miteinander addiert und das Ergebnis durch 43 dividiert. Der Rest beziehungsweise das Zeichen mit dem Wert des Restes ist dann die Prüfziffer.

Bei dem Text „BASICPRO“ sieht das folgendermaßen aus:

11 + 10 + 28 + 18 + 12 + 25 + 27 + 24 = 155
 155 / 43 = 3 Rest 26

Die Prüfziffer wäre also das Zeichen Nummer 26, das Q.

Anfangs-/Endezeichen

Anfangs- und Endezeichen des Code39 bestehen aus dem Bitmuster 100101101010. Einige Quellen bezeichnen dieses Bitmuster auch als *-Zeichen. Das ist allerdings verwirrend, da es zu der Annahme verleitet, der Stern

Ziffern	Bitmuster	Ziffern	Bitmuster	Ziffern	Bitmuster
00	10101100110010	34	11011010010100	68	10011011010010
01	10010110110100	35	11001101001010	69	10110011010010
02	10100110110100	36	11011001001010	70	10101001100110
03	10010011011010	37	11011010100100	71	10010101101100
04	10101100110100	38	11001101010010	72	10100101101100
05	10010110011010	39	11011001010010	73	10010010110110
06	10100110011010	40	10101100100110	74	10101001101100
07	10101101100100	41	10010110101100	75	10010100110110
08	10010110110010	42	10100110101100	76	10100100110110
09	10100110110010	43	10010011010110	77	10101011001100
10	11010100100110	44	10101100101100	78	10010101100110
11	11001010101100	45	10010110010110	79	10100101100110
12	11010010101100	46	10100110010110	80	11010100110010
13	11001001010110	47	10101101001100	81	11001010110100
14	11010100101100	48	10010110100110	82	11010010110100
15	11001001010110	49	10100110100110	83	11001001011010
16	11010010010110	50	11010110010010	84	11010100110100
17	11010101001100	51	11001011010100	85	11001010011010
18	11001010100110	52	11010011010100	86	11010010011010
19	11010010100110	53	11001001101010	87	11010101100100
20	10110100100110	54	11010110010100	88	11001010110010
21	10011010101100	55	11001011001010	89	11010010110010
22	10110010101100	56	11010011001010	90	10110100110010
23	10011001010110	57	11010110100100	91	10011010110100
24	10110100101100	58	11001011010010	92	10110010110100
25	10011010010110	59	11010011010010	93	10011001011010
26	10110010010110	60	10110110010010	94	10110100110100
27	10110101001100	61	10011011010100	95	10011010011010
28	10011010100110	62	10110011010100	96	10110010011010
29	10110010100110	63	10011001101010	97	10110101100100
30	11011010010010	64	10110110010100	98	10011010110010
31	11001101010100	65	10011011001010	99	10110010110010

Tabelle 7: Bitmuster für die Zeichendarstellung des Code25 Interleaved mit seinen breiten und schmalen Strichen.

Zeichen	Kombination	Wert	Zeichen	Kombination	Wert
0	101001101101	0	M	110110101001	22
1	110100101011	1	N	101011010011	23
2	101100101011	2	O	110101101001	24
3	110110010101	3	P	101101101001	25
4	101001101011	4	Q	101010110011	26
5	110100110101	5	R	110101011001	27
6	101100110101	6	S	101101011001	28
7	101001011011	7	T	101001001101	29
8	110100101101	8	U	110010101011	30
9	101100101101	9	V	100110101011	31
A	110101001011	10	W	110011010101	32
B	101101001011	11	X	100101101011	33
C	110110100101	12	Y	110010110101	34
D	101011001011	13	Z	100110110101	35
E	110101100101	14	-	100101011011	36
F	101101100101	15	.	110010101101	37
G	101010011011	16	Leerzeichen	100110101101	38
H	110101001101	17	\$	100100100101	39
I	101101001101	18	/	100100101001	40
J	101011001101	19	+	100101001001	41
K	110101010011	20	%	101001001001	42
L	101101010011	21			

Tabelle 8:
Bitmuster für die Zeichendarstellung des Code39.

sei ein vom Code39 darstellbares Zeichen. Das ist aber *nicht* der Fall.

Da auch das Endezeichen mit einem schwarzen Strich beginnt, muss zwischen dem letzten Zeichen des Textes und dem Endezeichen ebenfalls ein Leerraum eingefügt werden.

Code 128

Der 1981 entwickelte Code128 (Abb. 9) verdrängt seit Beginn der 90er Jahre den Code39 immer mehr. Das mag daran liegen, dass er vielseitiger ist und oft weniger Platz beansprucht. Eine der Ursachen ist aber sicherlich auch, dass er von EAN und UPC gewissermaßen adoptiert wurde (mehr dazu weiter unten).

Aufbau

Der Code128 besteht aus 107 verschiedenen Bitmustern, wobei die Muster 0 bis 102 jeweils drei verschiedene Werte darstellen können (siehe Tabelle 9).

Als Code A werden alle Ziffern, alle Großbuchstaben, sowie viele Sonder- und Steuerzeichen gedruckt. Beim Code B sind die ersten 64 Zeichen identisch mit Code A, aber an Stelle der Steuerzeichen können hier alle Kleinbuchstaben dargestellt werden. Die ersten 100 Zeichen in Code C stellen rein numerische Werte dar, wobei ähnlich dem Code25 Interleaved (siehe oben) immer zwei Ziffern als ein Zeichen ausgegeben werden.



Abbildung 9:
Der Code128 verdrängt seit Anfang der 90er Jahre zunehmend den Code39.

Welcher der drei Codes zum Tragen kommt, ist abhängig vom Startzeichen. Das Zeichen 103 am Beginn des Strichcodes zeigt an, dass das Folgende als Code A zu interpretieren ist. Bei den Startzeichen 104 und 105 gilt Entsprechendes für Code B und Code C.

Das Endezeichen ist immer das Zeichen Nummer 106.

Auch innerhalb des Strichcodes ist es möglich, auf einen anderen Code zu wechseln. Dazu dienen die Zeichen, die in Tabelle 9 als „Code A“ (B,C) bezeichnet sind. Um *nur für ein einziges Zeichen* den Code zu wechseln, gibt es das Zeichen „Shift“. Innerhalb von Code A eingesetzt, zeigt es an, dass das folgende Zeichen als Code B-Zeichen zu betrachten ist, und umgekehrt.

Prüfzeichen

Für den Code128 ist eine Prüfziffer, oder besser gesagt ein Prüfzeichen, zwingend erforderlich. Um es zu bestimmen, muss zunächst der Wert jedes Zeichens mit seiner Position im String multipliziert werden. Diese Produkte sind zum Wert des Startzeichens zu addieren. Das Ergebnis dieser Rechnung muss nun noch durch 103 dividiert werden, und der dabei verbleibende Rest ist dann der Wert des Prüfzeichens.

Das Beispiel sei wieder einmal der Text „basicpro“, diesmal jedoch kleingeschrieben. Wir berechnen also:

$$\text{„Start B“} + 1 * \text{„b“} + 2 * \text{„a“} + 3 * \text{„s“} \text{ usw.}$$

$$104 + 1*66 + 2*65 + 3*83 + 4*73 + 5*67 + 6*80 + 7*82 + 8*79 = 2862$$

$$2862 / 103 = 27 \text{ Rest } 81$$

Das Prüfzeichen hat also den Wert 81, und das ist im Code B das q.

EAN128

Neben dem „normalen“ Code128 gibt es seit einiger Zeit auch den so genannten EAN128. Er

dient dazu, Angaben zu normen, die in einem Code128 ausgegeben werden. Jeder EAN128 beinhaltet direkt hinter dem Startzeichen das Zeichen Nummer 102 („FNC 1“). Dahinter folgt als zwei- oder dreistellige Ziffernfolge der so genannte Datenbezeichner. Dieser gibt an, um welche Information es sich bei diesem Strichcode handelt. Der Datenbezeichner 10 zum Beispiel zeigt, dass in diesem Strichcode Angaben zu einer Chargennummer gemacht werden, bei der 17 handelt es sich um ein Verfallsdatum, usw. Der weitere Aufbau des Strichcodes (Anzahl und Inhalt der Stellen) kann unterschiedlich sein, abhängig vom Datenbezeichner.

Zurzeit sind bereits mehr als 50 solcher Datenbezeichner definiert. Eine Auswahl finden Sie in *Tabelle 10*.

Die Beispielprogramme auf der Heft-CD

Auf der Heft-CD befinden sich zwei Projekte, die den Umgang mit Strichcodes aus VB heraus zeigen sollen.

1. *bp_barcode.vbp* ist eine ActiveX-DLL, die zu allen in diesem Artikel beschriebenen Codes die Prüfziffer ermittelt und die Strichcodes wahlweise auf einem Drucker oder in einer PictureBox ausgibt.
2. *barcodetest.vbp* besteht nur aus einem Formular (Abbildung 10), mit dessen Hilfe diese ActiveX-DLL mit Parametern gefüttert wird.

ingenieurbüro drahn
software-entwicklung •••••

SOFTWARE ENTWICKLUNG

Dienstleistungsangebot

Individuelle SW-Entwicklung
Projektmitarbeit (auch vor Ort)
Entwicklung von Tools
Software-Design
Beratung

Programmiersprachen

Visual Basic
C/C++
Microsoft Access
Assembler
HTML, VBA

Standards

MFC
ADO/DAO
COM/DCOM
ActiveX

Kontakt

IBD – Ingenieurbüro Drahn
Melkendorfer Straße 19
D-95326 Kulmbach
Tel: 09221-97658 / Fax: -97659
e-mail: kontakt@ibd-software.de

www.ibd-software.de

Beide Programme sind absichtlich sehr simpel gehalten. So wurde bewusst auf jede Form des Error-Handlings verzichtet. Außerdem enthält *bp_barcode.vbp* keinerlei logische Prüfungen, zum Beispiel, ob der erfasste String tatsächlich als Strichcode druckbar ist.

Natürlich sind solche Prüfungen in einem professionellen Programm unverzichtbar. Wenn sie hier vernachlässigt worden sind, dann nur, weil dieses ActiveX nicht der großen Zahl von auf dem Markt befindlichen OCXen ein weiteres hinzufügen soll. Es soll nur zeigen, wie relativ einfach es ist, Barcodes aus VB heraus selbst zu drucken. Darum wurde auf alles verzichtet, was sich nicht unmittelbar auf dieses Thema bezieht.

Die Komponente enthält zwei Methoden: *Prüfziffer_berechnen* und *Drucken*. Dazu kommt noch die Eigenschaft *objAusgabe*. Vom Programm *BarCodeTest* aus wird sie initialisiert durch

```
Set Klasse = CreateObject("BP_BarCode.clsbp_barcode")
```

Ausgabemedium setzen

Bei *objAusgabe* handelt es sich um das Objekt, in das der Ausdruck erfolgen soll. Es muss vor dem Drucken aus dem aufrufenden Programm heraus gefüllt werden. Vom Programm *BarCodeTest* aus geschieht das durch den Code

```
Set Klasse.objAusgabe = Printer  
oder
```

```
Set Klasse.objAusgabe = Picture1
```

Hier kann jedes Objekt eingesetzt werden, das die Methode *Line* kennt.

Prüfziffer berechnen

An die Methode *Prüfziffer_berechnen* werden zwei Variablen übergeben. *intSchalter* ist ein Wert vom Typ *Integer*, der besagt, für welche Art von Strichcode hier die Prüfziffer ermittelt werden soll (1 = EAN13, 2 = UPC-A usw.). *strZeichenfolge* ist der String, für den die Prüfziffer zu berechnen ist.

Barcode drucken

Die Methode *Drucken* benötigt zusätzlich vier Variablen vom Typ *Integer*. Mit ihnen wer-

Wert	Code A	Code B	Code C	Kombination
000	SPACE	SPACE	00	11011001100
001	!	!	01	11001101100
002	„	„	02	11001100110
003	#	#	03	10010011000
004	\$	\$	04	10010001100
005	%	%	05	10001001100
006	&	&	06	10011001000
007	'	'	07	10011000100
008	((08	10001100100
009))	09	11001001000
010	*	*	10	11001000100
011	+	+	11	11000100100
012	,	,	12	10110011100
013	-	-	13	10011011100
014	.	.	14	10011001110
015	/	/	15	10111001100
016	0	0	16	10011101100
017	1	1	17	10011100110
018	2	2	18	11001110010
019	3	3	19	11001011100
020	4	4	20	11001001110
021	5	5	21	11011100100
022	6	6	22	11001110100
023	7	7	23	11101101110
024	8	8	24	11101001100
025	9	9	25	11100101100
026	:	:	26	11100100110
027	;	;	27	11101100100
028	<	<	28	11100110100
029	=	=	29	11100110010
030	>	>	30	11011011000
031	?	?	31	11011000110
032	@	@	32	11000110110
033	A	A	33	10100011000
034	B	B	34	10001011000
035	C	C	35	10001000110
036	D	D	36	10110001000
037	E	E	37	10001101000
038	F	F	38	10001100010
039	G	G	39	11010001000
040	H	H	40	11000101000
041	I	I	41	11000100010
042	J	J	42	10110111000
043	K	K	43	10110001110
044	L	L	44	10001101110
045	M	M	45	10111011000
046	N	N	46	10111000110
047	O	O	47	10001110110
048	P	P	48	11101110110
049	Q	Q	49	11010001110
050	R	R	50	11000101110
051	S	S	51	11011101000
052	T	T	52	11011100010
053	U	U	53	11011101110

Wert	Code A	Code B	Code C	Kombination
054	V	V	54	11101011000
055	W	W	55	11101000110
056	X	X	56	11100010110
057	Y	Y	57	11101101000
058	Z	Z	58	11101100010
059	[[59	11100011010
060	\	\	60	11101111010
061]]	61	11001000010
062	^	^	62	11110001010
063	_	_	63	10100110000
064	NU	`	64	10100001100
065	SH	A	65	10010110000
066	SX	B	66	10010000110
067	EX	C	67	10000101100
068	ET	D	68	10000100110
069	EQ	E	69	10110010000
070	AK	F	70	10110000100
071	BL	G	71	10011010000
072	BS	H	72	10011000010
073	HT	I	73	10000110100
074	LF	J	74	10000110010
075	VT	K	75	11000010010
076	FF	L	76	11001010000
077	CR	M	77	11110111010
078	SO	N	78	11000010100
079	SI	O	79	10001111010
080	DL	P	80	10100111100
081	D1	Q	81	10010111100
082	D2	R	82	10010011110
083	D3	S	83	10111100100
084	D4	T	84	10011110100
085	NK	U	85	10011110010
086	SY	V	86	11110100100
087	EB	W	87	11110010100
088	CN	X	88	11110010010
089	EM	Y	89	11011011110
090	SB	Z	90	11011110110
091	EC	{	91	11110110110
092	FS		92	10101111000
093	GS	}	93	10100011110
094	RS	~	94	10001011110
095	US	DEL	95	10111101000
096	FNC 3	FNC 3	96	10111100010
097	FNC 2	FNC 2	97	11110101000
098	SHIFT	SHIFT	98	11110100010
099	CODE C	CODE C	99	10111011110
100	CODE B	FNC 4	CODE B	10111101110
101	FNC 4	CODE A	CODE A	11101011110
102	FNC 1	FNC 1	FNC 1	11110101110
103	Start A	Start A	Start A	11010000100
104	Start B	Start B	Start B	11010010000
105	Start C	Start C	Start C	11010011100
106	Ende	Ende	Ende	1100011101011

Tabelle 9: Bitmuster für die Zeichendarstellung des Code128.

Datenbezeichner	Inhalt
00	Nummer der Versandeinheit
01	EAN der Handelseinheit
10	Chargennummer
11	Herstellungsdatum
15	Mindesthaltbarkeitsdatum
17	Verfallsdatum
20	Produktvariante
21	Seriennummer
30	Menge (enthaltene Stückzahl)
400	Bestellnummer des Warenempfängers
410	„Lieferung an“, ILN des Warenempfängers
421	„Lieferung nach“, Postleitzahl mit vorangestelltem 3-stelligem ISO-Ländercode

Tabelle 10:
Im EAN128 geben Datenbezeichner an, welche Art von Information der Strichcode enthält.

den Breite und Höhe des auszugebenden Strichcodes sowie die X- und Y-Positionen für die linke obere Ecke des Strichmusters definiert. Diese vier Werte werden in Millimetern angegeben. Die tatsächliche Breite des Ausdrucks kann ein wenig vom angegebenen Wert abweichen. Aus diesem wird nämlich zunächst die Breite eines einzelnen Striches in Twips ermittelt (567 Twips ergeben einen Zentimeter). Durch die dabei notwendige Rundung kann es zu geringfügigen Differenzen zwischen der gewünschten und der letztlich gedruckten Breite kommen.

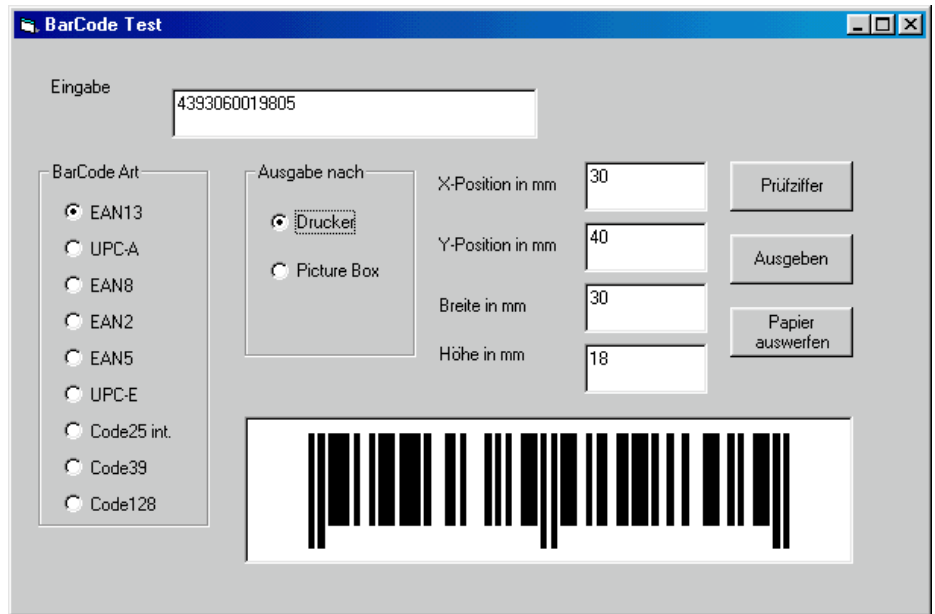


Abbildung 10:
Das Beispielprogramm BarCodeTest demonstriert die Anwendung der Strichcode-Komponente bp_barcode.vbp. Mit ihr können Sie Barcodes in eine PictureBox oder auf den Drucker ausgeben.

Die Zukunft hat bereits begonnen

Ursprünglich steckte hinter der Entwicklung von Strichcodes die Idee, dem Computer das Lesen beizubringen. Wie die Beispiele oben

zeigen, konnte das erfolgreich realisiert werden. Inzwischen geht die Entwicklung dahin, möglichst viele Informationen auf möglichst kleinem Raum unterzubringen. So wird darauf hingearbeitet, Texte von mehr als 10.000 Zeichen (!) auf der Fläche eines normalen

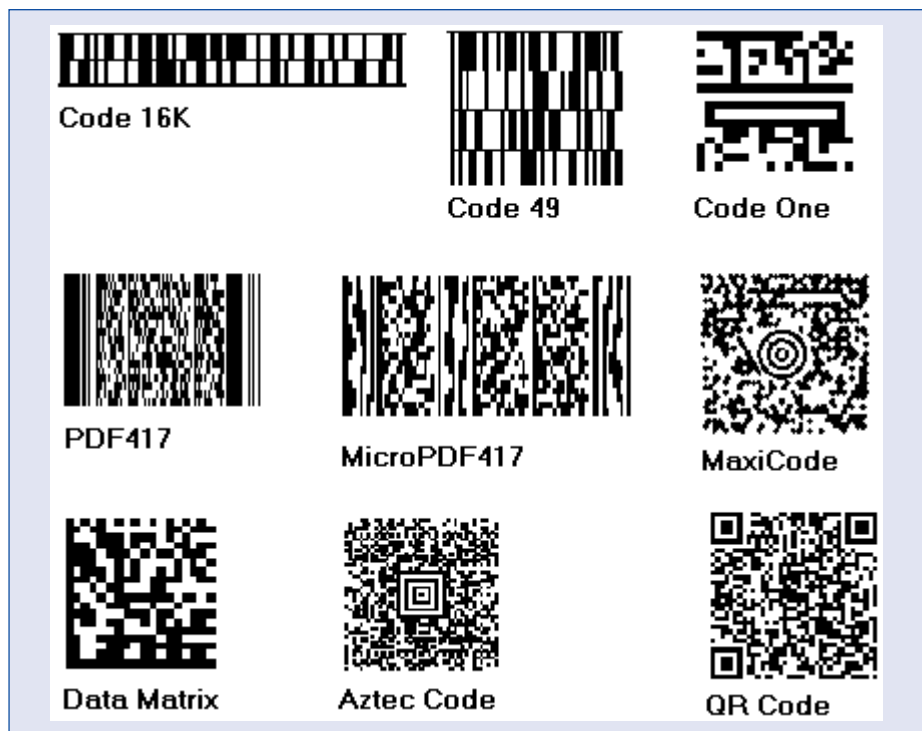


Abbildung 11: Beispiele für moderne optische Codes. Sie unterscheiden sich in Informationsdichte und Leserichtung enorm von den bisherigen Strichcodes.

Passbildes unterzubringen. Dass diese Texte von einem menschlichen Auge nicht mehr gelesen werden können, ist ein (möglicherweise gewünschter) Nebeneffekt. Diese Codes haben auch optisch kaum noch etwas mit den bekannten „Strich“-Codes gemein. Es handelt sich um Bitmuster, die gleichzeitig waagrecht *und* senkrecht, teilweise auch diagonal oder sogar spiralförmig gelesen werden. Einige Beispiele für solche optischen Codes, die bereits existieren, sind in *Abbildung 11* dargestellt.

Welcher Code ist der beste?

Die Frage, welcher der beschriebenen Codes für eine Anwendung am besten ist, lässt sich

nicht ohne Weiteres beantworten. In der Regel hängt die Entscheidung davon ab, wo der Code gelesen werden soll.

Im Bereich der rein numerischen Strichcodes wird in den meisten Fällen wohl der EAN13 zur Anwendung kommen. Dieser Code kann von den meisten Lesegeräten, sei es ein Scanner an der Kasse oder ein Lesestift an der PC-Tastatur, problemlos erkannt werden. Das gilt auch für preiswerte und ältere Geräte. Hinzu kommt, dass es sich häufig darum handeln wird, Artikelnummern einzulesen, die von den Herstellern nach EAN-Richtlinien vergeben wurden. In diesem Fall ist es sicher sinnvoll, wenn der Strichcode, der auf dem Lieferschein gedruckt wird, identisch ist mit dem, den der Hersteller schon auf der Verpackung angibt. Der EAN13-

Code ist somit nicht nur ein Code für die Umsetzung von Ziffern in Striche, sondern zielt auf einen bestimmten Zweck – Kodierung von Artikelnummern – und kommt mit einer Dienstleistung daher (Vergabe eindeutiger Artikelnummern).

Was im europäischen Bereich für den EAN-Code gilt, lässt sich für den amerikanischen beziehungsweise kanadischen Markt auf den UPC-Code übertragen.

Bei den alphanumerischen Strichcodes wird wohl dem Code128 die Zukunft gehören. Wie bereits weiter oben erwähnt, verdrängt er mehr und mehr die anderen Codes. Generell muss aber auch hier die Entscheidung für einen bestimmten Barcode in erster Linie davon abhängig gemacht werden, dass dieser von allen eingesetzten Lesegeräten erkannt werden kann. ▼

Ressourcen

Bei der Recherche zum Thema Strichcodes stößt man im Internet auf eine unüberschaubare Zahl von Links. Die meisten führen zu Anbietern von Barcode-Lesegeräten oder -Druckern, andere zu Software, mit der man Strichcodes drucken kann. Auch BasicPro hat sich dieses Themas bereits angenommen und „ActiveBarcode“ von Schenk & Horn als Tool des Monats der Ausgabe 3/99 beigelegt.

Verweise zu grundlegenden Informationen über Strichcodes sind demgegenüber vergleichsweise rar. Die nachstehende Auswahl soll Ihnen den Einstieg in die eigene Recherche erleichtern.

<http://www.barcodeusa.com/barcode.htm>

http://www.semiconductor.agilent.com/barcode/sg/Misc/bc_ref.html

<http://www.adams1.com/pub/russadam/info.html#Specs>

<http://www.herdsoft.com/ti/barvis/bvident.html>

<http://www.herdsoft.com/ti/barvis/bvleit.html>

<http://www.herdsoft.com/ti/barvis/bvean.html>

<http://www.strichcode.com/index.html>

Seine ersten Schritte in der Programmierung machte Stephan Kerbei auf einem Commodore PET 2001, also noch einige Generationen vor dem legendären VC64. 1983 konnte er sein Hobby zum Beruf machen. Seitdem erstellt er Anwendungen im kaufmännischen Bereich für einige Großunternehmen in COBOL und seit einiger Zeit auch immer häufiger in VB. Für Fragen und Anregungen erreichen Sie Stephan Kerbei per EMail an StephanK@basicpro.de.