Čárové kódy EAN v TEXu

Petr Olšák

Z originálu napsaného v kostrbaté angličtině (článek určený pro TUGboat) pokud možno věrně přeložil Petr Olšák. Důsledkem je poněkud neohrabaná čeština. Věřím, že text přitom neztratil na srozumitelnosti a že i přes tyto „jazykové problémy" bude pro čtenáře užitečný.

Abstrakt. V tomto článku popíšeme algoritmus na transformaci kódu EAN 13 ve tvaru třináctimístného čísla do čárového kódu (řady různě velkých čar a mezer). Ukážeme implementaci tohoto algoritmu v makrojazyku TEXu. Kreslení čar je realizováno pomocí TEX primitivu `\vrule`. Dále jsou ukázány některé údaje z norem (například o tolerancích). Makro diskutované níže je k dispozici na CTANu.

Když jsem připravoval svou první knihu o TEXu [3], můj zájem o přípravu této knihy nekončil odevzdáním rukopisu nebo předlohy se sazbou v nějakém nakladatelství. Protože nakladatelem je CSTUG, stal jsem se prakticky sám sobě redaktorem. Dělal jsem si návrh obálky, řešil jsem problémy následné distribuce, získání ISBN atd.

Jakmile jsem získal ISBN (International Standard Book Number), konvertoval jsem toto číslo do EAN 13 (European Article Numbering) a začal jsem se starat o to, jak získat k tomuto číslu čárový kód, protože tento kód bývá obvykle na zadních stranách obálek knížek. Shledal jsem, že ceny pořízení čárového kódu u komerčních firem jsou velmi vysoké. Na druhé straně vytvoření čárového kódu TEXem je velmi přirozená aplikace tohoto programu díky jeho vysoké přesnosti a algoritmickému makrojazyku. Jediný problém bylo sehnat popis algoritmu, který třináctimístná čísla na vstupu převádí na metriku čar a mezer na výstupu. Tento algoritmus jsem nakonec našel v [1].

Transformace z ISBN do EAN je jednoduchá. ISBN je desetimístné číslo. Pomlčky uvnitř čísla je rozdělují na pole (země-nakladatel-číslo-kontrola) a můžeme tyto pomlčky ignorovat. Na začátek čísla přidáme novou konstatní trojici číslic (978). Dále odebereme poslední kontrolní číslici a vypočítáme ji znovu. Algoritmus na výpočet kontrolní číslice pro ISBN je odlišný od algoritmu používaného u EAN. Pro EAN nejprve spočítáme součet číslic na sudých pozicích. Označímej jej e. Pak spočítáme součet číslic na lichých pozicích (bez kontrolní číslice). Takový součet označíme o. Dále vypočteme $3 \times e + o$. Rozdíl tohoto výsledku od nejbližšího násobku deseti je kontrolní číslice. Například TEXbook s pevnou obálkou [2] má ISBN 0-201-13447-0 (0: země USA, 201: nakladatel Addison Wesley, 13447: číslo knihy přidělené nakladatelem, 0: kontrolní číslice). Po přidání tří konstatních číslic na začátek a odstranění kontrolní číslice dostaneme „987020113447?". Nyní $e = 7 + 0 + 0 + 1 + 4 + 7 = 19$ a $o = 9 + 8 + 2 + 1 + 3 + 4 = 27$. Rozdíl mezi $3 \times 19 + 27 = 84$ a 90 je 6, což je kontrolní číslice. Výsledné EAN můžeme rozdělit pomocí pomlček na pole po šesti číslicích (jen z důvodu snadnějšího čtení) a dostáváme 9-780201-134476.

Transformace z čísla EAN do čárového kódu je poněkud komplikovanější. První číslice je pro knihy 9, nicméně pro ostatní druhy zboží může být jiná. Tato číslice nemá své vlastní pole v čárovém kódu, ale ovlivní algoritmus transformace ostatních číslic do svých polí v čárovém kódu. Šířka čáry nebo mezera mezi čarami v kódu je vyjádřena v násobcích základní jednotky zvané „modul X". Velikost tohoto modulu může být podle různých předpisů v normě různá (viz SC předpisy níže), ale základní rozměr je 0,33 mm. Každá číslice z pozic 12 až 1 je transformována do pole o šířce 7X (sedm modulů). Toto pole obsahuje vždy dvě čáry a dvě mezery. Dále je na začátek kódu připojen tzv. „startovní znak" o šířce 3X (1X čára, 1X mezera a 1X čára). Stejný znak je připojen jako tzv. „koncový znak" na konec kódu a doprostřed kódu mezi 7. a 6. číslici je vložen tzv. „oddělovací znak" o šířce 5X (1X mezera, 1X čára, 1X mezera, 1X čára a 1X mezera). Tyto speciální znaky mají čáry poněkud delší a přesahují přes dolní okraj o 5X.

Snadno zjistíme, že celková délka EAN kódu je 95X. Dále musíme připočíst bílé místo o šířce 11X vlevo od kódu a 7X vpravo od kódu. Toto je minimální bílý okraj, který je nutno dodržet pro kódy umístěné na barevném pozadí. Celkový počet čar v kódu je 30.

Každá číslice je transformována do dvou čar v poli o šířce 7X pomocí jedné ze tří následujících tabulek (A, B a C).

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

Tabulky A, B a C.

Nula v tabulce znamená bílý modul (šířky X) a jednička označuje černý modul. Například číslice 4 je konvertována podle tabulky A na řadu 1X mezera, 1X čára, 3X mezera a 2X čára a podle tabulky B na řadu 2X mezera, 3X čára, 1X mezera a 1X čára. Všimneme si, že skutečně všechny tabulky konvertují číslice na právě dvě čáry a dvě mezery a že příslušné pole začíná v případě tabulek A a B mezerou a v případě tabulky C čarou.

Číslice z pozic 6 až 1 se tranformují vždy podle tabulky C. Transformace číslic na pozicích 12 až 7 se provede podle tabulky A nebo B. Výběr je podmíněn hodnotou třinácté číslice (první zleva) podle tabulky „Závislost na třinácté číslici“.

13. číslice	12	11	10	9	8	7
0	A	A	A	A	A	A
1	A	A	B	A	B	B
2	A	A	B	B	A	B
3	A	A	B	B	B	A
4	A	B	A	A	B	B
5	A	B	B	A	A	B
6	A	B	B	B	A	A
7	A	B	A	B	A	B
8	A	B	A	B	B	A
9	A	B	B	A	B	A

Závislost na třinácté číslici.

Například, pokud je na třinácté pozici číslice 9 (náš případ s knihami) pak jsou podle tabulky A transformovány číslice z pozic 12, 9 a 7 a podle tabulky B číslice z pozic 11, 10 a 8.

Nyní ukážeme TeXovské makro. Nejprve zavedeme speciální font OCRb pro zapsání EAN v čitelné podobě pod čárovým kódem. METAFONTové zdroje těchto fontů jsou k dispozici na CTANu. Vytvořil je Norbert Schwarz. Udělal jsem v těchto zdrojích jeden drobný zásah – přidal jsem na začátek souboru `ocrbmac.mf` příkaz „`mode_setup;`“.

```
\message{The EAN-13 barcodes macro. Copyright (C) Petr Olsak, 1995}
\font\ocrb=ocrb9       % for EAN in ``number form''
\font\ocrbsmall=ocrb7  % for ISBN
```

Nyní deklarujeme některé „proměnné“:

```
\newcount\numlines \newcount\nummodules  % number of bars and of modules.
\newcount\numdigit \newcount\evensum \newcount\oddsum  % internal variables
\newdimen\X            % the module size X,
\newdimen\bcorr        % the bar correction (see bellow).
\newdimen\workdimen \newdimen\barheight   % internal variables
```

Hlavní makro `\EAN` konvertuje třináctimístné číslo EAN na vnitřní reprezanci kódu jako šedesátimístné číslo `\internalcode`. Každá číslice z `\internalcode` reprezentuje násobek modulu X pro čáru nebo mezeru. Číslice jsou napsány ve stejném pořadí, jako je pořadí mezer a čar v kódu. Liché pozice (počítáno zleva) v `\internalcode` reprezentují mezery a sudé čáry.

Makra použijeme například takto: `\EAN 9-780201-134476`. Přítomnost znaků „-“ nemá vliv. Makro čte 13 číslic a uloží je do `\firstdigit`, `\frontdigits` a `\enddigits`. Současně makro konvertuje vstup do `\internalcode` použitím `\settables`, `\usetabAB`, `\insertseparator` a `\usetabC`.

```
\def\internalcode{0111}              % Begin mark at start
\def\frontdigits{}                    % 12--7 digit of EAN
\def\EAN{\begingroup\EANscan}
\def\EANscan#1{\if#1-\let\next=\EANscan \else
    \advance\numdigit by1
    \ifnum\numdigit<13
      \ifodd\numdigit \advance\oddsum by #1 \else \advance\evensum by #1 \fi
      \let\next=\EANscan
      \ifnum\numdigit=1 \settables#1\def\firstdigit{#1}\else
      \ifnum\numdigit<8 \usetabAB#1\edef\frontdigits{\frontdigits#1}\else
      \ifnum\numdigit=8 \insertseparator \A \usetabC #1\def\enddigits{#1}%
      \else             \usetabC#1\edef\enddigits{\enddigits#1}%
      \fi\fi\fi
    \else \testchecksum#1\usetabC#1\edef\enddigits{\enddigits#1}%
      \let\next=\EANclose
    \fi\fi \next}
```

Makro `\testchecksum` kontroluje správnost poslední (kontrolní) číslice kódu EAN.

```
25 \def\testchecksum#1{\multiply\evensum by3 \advance\evensum by\oddsum
26     \oddsum=\evensum
27     \divide\oddsum by10 \multiply\oddsum by10 \advance\oddsum by10
28     \advance\oddsum by-\evensum \ifnum\oddsum=10 \oddsum=0 \fi
29     \ifnum#1=\oddsum \else
30     \errmessage{The checksum digit has to be \the\oddsum, no #1 !}\fi}
```

Při expanzi makra \EANclose se uzavře \internalcode pomocí \insertendmark. Dále se napíše do log souboru přečtené číslo EAN v třináctimístné formě a též v jeho šedesátimístné vnitřní reprezentaci. Poslední aktivitou je „spuštění“ makra \EANbox, které vytvoří box s čarami a mezerami. Vstupním parametrem pro toto makro je šedesátimístné \internalcode.

```
31 \def\EANclose{\insertendmark
32     \wlog{EAN: \firstdigit\space\frontdigits\space\enddigits}%
33     \wlog{EANinternal: \internalcode}%
34     \expandafter\EANbox\internalcode..\endgroup}
```

Jak je vytvářena vnitřní reprezentace \internalcode? Na tuto otázku dávají odpověď následující makra. V těchto makrech poznáváme tabulky zmíněné výše, které jsou přepsány do makrojazyka TEXu.

```
35 \def\A{\def\0{3211}\def\1{2221}\def\2{2122}\def\3{1411}\def\4{1132}%
36   \def\5{1231}\def\6{1114}\def\7{1312}\def\8{1213}\def\9{3112}}
37 \def\B{\def\0{1123}\def\1{1222}\def\2{2212}\def\3{1141}\def\4{2311}%
38   \def\5{1321}\def\6{4111}\def\7{2131}\def\8{3121}\def\9{2113}}
39 \def\settables#1{\ifnum#1=0 \def\tabs{\A\A\A\A\A\A}\fi
40                  \ifnum#1=1 \def\tabs{\A\A\B\A\B\B}\fi
41                  \ifnum#1=2 \def\tabs{\A\A\B\B\A\B}\fi
42                  \ifnum#1=3 \def\tabs{\A\A\B\B\B\A}\fi
43                  \ifnum#1=4 \def\tabs{\A\B\A\A\B\B}\fi
44                  \ifnum#1=5 \def\tabs{\A\B\B\A\A\B}\fi
45                  \ifnum#1=6 \def\tabs{\A\B\B\B\A\A}\fi
46                  \ifnum#1=7 \def\tabs{\A\B\A\B\A\B}\fi
47                  \ifnum#1=8 \def\tabs{\A\B\A\B\B\A}\fi
48                  \ifnum#1=9 \def\tabs{\A\B\B\A\B\A}\fi}
49 \def\usetabAB#1{\expandafter\scantab\tabs\end \usetabC #1}
50 \def\scantab#1#2\end{#1\def\tabs{#2}} % The tab #1 is activated and removed
51 \def\usetabC#1{\edef\internalcode{\internalcode\csname#1\endcsname}}
52 \def\insertseparator{\edef\internalcode{\internalcode 11111}}
53 \def\insertendmark{\edef\internalcode{\internalcode 111}}
```

Není potřeba definovat tabulku C, protože tato tabulka je přesně „negativní“ k tabulce A. Vložením oddělovacího znaku (řádek 52 v ukázce) se přidá do \internalcode lichý počet číslic (jmenovitě 5). Z toho plyne, že se převrátí význam mezery a čáry pro další číslice. Použití tabulky \A je pak pro další transformace číslic z pozic 6 až 1 dostačující.

Nyní přichází nejdůležitější část našeho makra – vytváření čar pomocí primitivu \vrule. Tuto práci dělá interní makro \EANbox. Makro postupně čte jako parametr šedesáticiferné číslo \internalcode ukončené dvěma tečkami. V každém kroku načte dvě číslice (první reprezentuje mezeru a druhé šířku čáry) a vloží odpovídající \kern a \vrule. Každá dvojice kern/rule je navíc korigována podle tzv. „čárové korekce“. Norma doporučuje udělat každou čáru poněkud užší, než vyplývá z přesného násobku modulu X. Toto doporučení vyplývá z vlastností inkoustu při tisku. Například pro ofsetovou technologii se doporučuje redukovat čáry o 0.020 mm. Při redukci šířek čar samozřejmě musíme zvětšit o stejnou hodnotu mezery, abychom udrželi stejnou vzdálenost mezi čarami.

Čáry 1, 2, 15, 16, 29 a 30 mají nenulovou „hloubku“ (5X), protože se jedná o čáry startovního, oddělovacího a koncového znaku. Za normálních okolností je výška čáry 69,24 X, ale tato výška může být redukována třeba v případě, že je nahoru ke kódu přidáno číslo ISBN. Jestliže je `\barheight` nulový, pak je použita implicitní výška. Jinak je použita výška čáry podle `\barheight`. Tato vlastnost dává uživateli možnost nastavit výšku čáry individuálně.

```
\def\EANbox{\vbox\bgroup\offinterlineskip
  \setbox0=\hbox\bgroup \kern11\X\EANrepeat}
\def\EANrepeat#1#2{\if#1.\let\next=\EANfinal \else\let\next=\EANrepeat
  \advance\numlines by1
  \advance\nummodules by#1 \advance\nummodules by#2
  \workdimen=#1\X \advance\workdimen by \bcorr \kern\workdimen
  \workdimen=#2\X \advance\workdimen by-\bcorr \vrule width\workdimen
     \ifdim\barheight=0pt height 69.24242424\X \else height\barheight \fi
     \ifnum\numlines=1    depth5\X\else  % the start mark
     \ifnum\numlines=2    depth5\X\else
     \ifnum\numlines=15   depth5\X\else  % the separator mark
     \ifnum\numlines=16   depth5\X\else
     \ifnum\numlines=29   depth5\X\else  % the end mark
     \ifnum\numlines=30   depth5\X\else depth0pt \fi\fi\fi\fi\fi\fi
  \fi\next}
```

Makro `\EANfinal` kontroluje správnost přečteného `\internalcode`. Počet číslic musí být 60 a součet číslic musí být 95 (protože je 95 modulů X v kódu celkem). Jestliže tyto podmínky nejsou splněny, je aktivováno makro `\internalerr`. Tato situace by neměla nikdy nastat. Tato chyba signalizuje, že je chybná nějaká interní tabulka nebo je porušena konzistence makra.

Makro `\EANfinal` sestaví `\vbox`. „Hloubka“ vnitřního `\hbox`u s kódem je 5X, protože taková je „hloubka“ čar pro startovní, oddělovací a koncový znak. Přepíšeme tuto hloubku na 0pt, abychom mohli přidat číslo EAN v čitelném tvaru těsně pod kód.

```
\def\EANfinal{\testconsistence
  \kern7\X\egroup
  \hbox{\ocrbsmall \kern10\X \ISBNnum}\kern1\X
  \dp0=0pt \box0 \kern-1\X
  \hbox{\ocrb\kern2\X\firstdigit\kern5\X \frontdigits\kern5\X \enddigits}
  \egroup \global\barheight=0pt \gdef\ISBNnum{}}
\def\testconsistence{\ifnum\numlines=30\else\internalerr\fi
  \ifnum\nummodules=95\else\internalerr\fi}
\def\internalerr{\errmassage{Sorry, my internal tables are wrong, may be.}}
```

Jestliže uživatel napíše ISBN pomocí makra `\ISBN` (například `\ISBN 0-201-13447-0 `), tento údaj se připojí nad kód a výška čar v kódu bude redukována.

```
\barheight=0pt
\def\ISBNnum{}
\def\ISBN #1 {\def\ISBNnum{ISBN #1}\barheight=45.151515\X\relax}
```

Konečně definujme velikost modulu `\X` a čárovou redukci.

```
\X=.33mm        % Basic size 100%, SC2 code
\bcorr=.020mm  % Bar-correction for offset process
\endinput
```

Makro je uloženo v souboru `ean13.tex` a bylo testováno v plainu následujícím způsobem:

```
\input ean13
\nopagenumbers
\ISBN 80-901950-0-8  \EAN 978-80-901950-0-4   % Typografický systém TeX
\end
```

Výsledek vypadá takto:

ISBN 80-901950-0-8

9 788090 195004

Makro pracuje rovněž v LaTeXu 2.09. LaTeX 2_ε nebyl testován.

Na konec tohoto článku srovnáme povolené tolerance popsané v normě, přesnost TeXu a možnosti některých výstupních zařízení.

Velikost modulu X může být různá. Makro uvedené výše vytváří čárové kódy EAN pro základní velikost modulu X 0,33 mm. Tato velikost je popsána v tzv. SC2 variantě normy jako základní velikost 100%. Norma umožňuje použít některé další velikosti modulu X. Změnou parametru `\X` v našem makru můžeme získat další velikosti kódu. Samozřejmě, že musíme též změnit velikost zavedeného fontu OCRb.

Možnosti povolených velikostí modulu X jsou popsány v tabulce „Různé velikosti modulu X".

Malé rozměry modulu X jsou doporučeny pro výstupní zařízení s vysokou kvalitou a velké velikosti X dávají možnost kreslit čárové kódy na zařízeních s nižším rozlišením.

Norma popisuje tři speciální toleranční parametry, které jsou závislé na velikosti modulu X. Parametr a specifikuje toleranci pro šířku čáry, parametr b udává toleranci pro vzdálenost mezi hranami (levá nebo pravá hrana ale vždy stejná) dvou sousedních čar a parametr c popisuje toleranci pro šířku pole odpovídající jedné číslici (tedy přesně šířku 7X). Následující tabulka „Tolerance" popisuje tyto tolerance v mikrometrech (μm). Nechápu, proč sloupec „vel. X" z předchozí tabulky přesně neodpovídá témuž sloupci „vel. X" z následující tabulky. Bohužel, normy často bývají tajemné.

Nyní můžeme začít srovnávat. Uvažujme základní velikost kódu 100% (modul X je 0,33 mm). Tolerance šířky čáry je 101 μm, (ne)přesnost TeXu je 0,0054 μm, velikost pixelu pro osvitovou jednotku při 2 400 dpi je zhruba 10 μm a doporučená čárová redukce pro ofsetovou technologii je 20 μm. Pokud nastavíme osvitovou jednotku na 1 200 dpi, je nepřesnost jejího výstupu srovnatelná s doporučenou čárovou redukcí pro ofset.

V závislosti na vnitřním algoritmu `dvi` ovladače se tedy vysoká přesnost TeXu ztrácí a některé toleranční parametry mohou způsobit těžkosti. Zaokrouhlovací algoritmy `dvi` ovladačů mohou pracovat dvěma různými způsoby. Za prvé, ovladač umístí každou jednotlivou čáru (rule) do výstupního rastru individuálně a zaokrouhlí až v okamžiku umístění. Podle druhého způsobu ovladač nejprve zaokrouhlí řadu kern, rule, kern, rule… na jednotky pixelů a pak už při umisťování čar pracuje jen s těmito jednotkami. Tento přístup může vést k akumulaci chyby a může způsobit potíže s parametrem c. Zdá se ale, že pro čtečky čárových kódů je tento druhý přístup mnohem přijatelnější, protože důležitější je dodržet rovnoměrné vzdálenosti mezi sousedními objekty, než dbát na dodržení parametrů pro globální šířku kódu. To je ale jen můj názor.

vel. X	norma	koef.	rozměr včetně okrajů
0,264	SC0	0,800	29,83 × 21,00
0,270	SC0	0,818	30,58 × 21,53
0,281	SC0	0,850	31,70 × 22,32
0,297	SC1	0,900	33,56 × 23,63
0,313	SC1	0,950	35,43 × 24,94
0,330	SC2	1,000	37,29 × 26,26
0,346	SC2	1,050	39,15 × 27,58
0,363	SC3	1,100	41,02 × 28,29
0,379	SC3	1,150	42,88 × 30,20
0,396	SC4	1,200	44,75 × 31,51
0,412	SC4	1,250	46,61 × 32,82
0,429	SC5	1,300	48,48 × 34,14
0,445	SC5	1,350	50,34 × 35,45
0,462	SC5	1,400	52,21 × 36,76
0,478	SC5	1,450	54,07 × 38,08
0,495	SC6	1,500	55,94 × 39,39
0,511	SC6	1,550	57,80 × 40,70
0,528	SC7	1,600	59,66 × 42,01
0,544	SC7	1,650	61,53 × 43,33
0,561	SC7	1,700	63,39 × 44,64
0,577	SC7	1,750	65,26 × 45,96
0,594	SC8	1,800	67,12 × 47,26
0,610	SC8	1,850	68,99 × 48,58
0,627	SC8	1,900	70,85 × 49,90
0,643	SC8	1,950	72,72 × 51,20
0,660	SC9	2,000	74,58 × 52,52
0,700	SC10	2,120	79,05 × 55,67

Různé velikosti modulu X.

Zjistil jsem, že `dvi` ovladače většinou zaokrouhlují šířku čáry (rule) na jednotky pixelů *nahoru* a nikdy ne *dolů*. Důsledkem toho je skutečnost, že mezery mají tendenci být o jeden pixel užší než šířka čar, které by podle pravidel měly mít stejnou velikost. Proto doporučuji přidat k čárové redukci (k registru `\bcorr`) ještě navíc polovinu velikosti pixelu výstupního zařízení.

Slyšel jsem, že čárové kódy EAN jsou úspěšně čteny i ze štítků vytištěných pomocí jehličkových tiskáren s velmi nízkým rozlišením. Přitom je použit základní rozměr 100% nebo jen nepatrně větší. Ve světle této skutečnosti můžeme prohlásit, že tolerance vstupních zařízení čárových kódů jsou obvykle podstatně větší, než tolernce předepsané normou.

Literatura

[1] Adriana Benadiková, Štefan Mada a Stanislav Weinlich. *Čárové kódy, automatická identifikace*. Grada 1994, 272 stran. ISBN 80-85623-66-8.
[2] Donald Knuth. *The TEXbook*, díl A z *Computers & Typesetting*. Addison-Wesley, Reading, MA, USA 1986, ix+483 stran. Pevná obálka ISBN 0-201-13447-0.
[3] Petr Olšák. *Typografický systém TEX*. CSTUG 1995, 270 stran. ISBN 80-901950-0-8.

vel. X	$\pm a$	$\pm b$	$\pm c$
0,26	32	38	75
0,28	52	41	81
0,30	72	44	87
0,32	92	47	93
0,33	101	49	96
0,34	105	50	99
0,36	115	53	104
0,38	124	56	110
0,40	134	59	116
0,42	143	62	122
0,44	152	65	128
0,46	162	68	133
0,48	171	71	139
0,50	181	73	145
0,52	190	76	151
0,54	199	79	157
0,56	209	82	162
0,58	218	85	168
0,60	228	88	174
0,62	237	91	180
0,64	246	94	186

Tolerance.