

Politechnika Wrocławska  
Wydział Informatyki i Telekomunikacji

## URZĄDZENIA PERYFERYJNE

# Sprawozdanie - Kody Kreskowe EAN-13

Studenci:

Julia Gościniak 259164

Maciej Radecki 253257

Data zajęć:

Piątek TN 13:15

Prowadzący:

dr inż. Jan Nikodem

6 listopada 2022

# Spis treści

<b>1</b>	<b>Wstęp</b>	<b>2</b>
1.1	Kody kreskowe . . . . .	2
1.1.1	Wymiarowość kodu . . . . .	2
1.1.2	Ciągłość kodu i zakres kodowania . . . . .	2
1.2	EAN . . . . .	2
1.2.1	Wymiary kodu EAN-13 . . . . .	2
1.2.2	Struktura kodu kreskowego EAN-13 . . . . .	2
1.2.3	Obliczanie cyfry kontrolnej kodu kreskowego EAN-13 . . . . .	3
1.3	Cel ćwiczenia . . . . .	3
<b>2</b>	<b>Realizacja ćwiczenia</b>	<b>3</b>
2.1	Działanie aplikacji . . . . .	3
2.2	Generowanie kodu kreskowego . . . . .	4
2.3	Implementacja aplikacji . . . . .	5
2.3.1	Klasa Program.cs . . . . .	5
2.3.2	Klasa Form.cs . . . . .	5
2.3.3	Klasa Ean.cs . . . . .	5
2.4	Najważniejsze funkcje programu . . . . .	5
<b>3</b>	<b>Podsumowanie i wnioski</b>	<b>6</b>
<b>4</b>	<b>Źródła</b>	<b>6</b>

# 1 Wstęp

## 1.1 Kody kreskowe

Kody kreskowe to graficzna reprezentacja informacji poprzez kombinację ciemnych i jasnych elementów, ustaloną według symboliki reguł opisujących budowę danego kodu.

### 1.1.1 Wymiarowość kodu

Wyróżnić można 3 główne kategorie kodów:

- jednowymiarowe - złożone najczęściej z grupy pionowych pasów o różnych szerokościach np. EAN-13
- dwuwymiarowe - dzielą się na piętrowe (kilka kodów jednowymiarowych jeden pod drugim) i matrycowe gdzie informacja zapisana jest na określonej powierzchni, z tym że do zapisu nie są wykorzystywane kreski, lecz inne oznaczenia np. prostokąty w kodach QR
- trójwymiarowe - zazwyczaj są to wytłaczane dowolne kody jednowymiarowe, różnica polega na tym, iż w zapisie/odczytzie zamiast różnic w kolorach wykorzystuje się różnice głębokości tłoczenia.

### 1.1.2 Ciągłość kodu i zakres kodowania

Kody kreskowe można podzielić na ciągłe (nie występują w nich przerwy między kodowanymi znakami) i nie ciągłe gdzie taka przerwa może wystąpić. Kodowane są albo wyłącznie cyfry albo znaki z kodu ASCII.

## 1.2 EAN

**European Article Number (EAN)** to standard opisujący kody kreskowe i ich numerowanie używane w międzynarodowym handlu. Numer opisuje typ produktu, pakowanie oraz producenta. Najpopularniejszym standardem jest EAN-13, który zawiera trzycyfrowy prefiks GS1 (oznacza on kraj rejestracji produktu lub specjalny typ produktu), kod producenta, kod produktu oraz cyfrę kontrolną.

### 1.2.1 Wymiary kodu EAN-13

Kod EAN-13 ma ściśle określone wymiary. Kod kreskowy EAN-13 można przeskalować ze współczynnikiem skalowania w zakresie  $[0,8; 2,0]$ . Skalowanie spowoduje powstanie kodu kreskowego między minimalnym dopuszczalnym rozmiarem 29,83 mm szerokości i 20,74 mm wysokości a maksymalnym dopuszczalnym rozmiarem 74,58 mm szerokości i 51,86 mm wysokości.

### 1.2.2 Struktura kodu kreskowego EAN-13

Kod kreskowy jest podzielony na 113 modułów jednakowej szerokości, z których każdy może być czarny (reprezentując logiczne 1) lub biały (reprezentując logiczne 0). Wskazując od lewej kolejne sekcje to:

- 11 modułów o wartości 0 przedstawiających "Quiet Zone", ułatwiająca odczyt kodu skanerom
- 3 moduły markera startu kodu o wartości (101)
- 42 moduły wskazujące wartość pierwszych 7 cyfr kodu kreskowego
- 5 modułów markera środka kodu o wartości (01010)
- 42 moduły wskazujące wartość ostatnich 6 cyfr kodu kreskowego
- 3 moduły markera stopu o wartości (101)
- 7 modułów o wartości 0 przedstawiających "Quiet Zone" z prawej strony

Aby zakodować 13 cyfrową liczbę, cyfry podzielone są na 3 grupy. W pierwszej grupie znajduje się jedynie pierwsza cyfra kodu. Druga grupa składa się z kolejnych 6 cyfr kodu i jest zakodowana używając wzoru oznaczonego literą A oraz B. Trzecia grupa składająca się z 6 ostatnich cyfr zakodowana jest używając wzoru oznaczonego literą C, która jest negacją kodu oznaczonego literą A. Pierwsza cyfra zakodowana jest poprzez użycie odpowiednich kodowań (A oraz B) dla kolejnych cyfr znajdujących się w drugiej grupie, których kolejność można zobaczyć na poniższej tabeli.

Struktura EAN-13:

Pierwsza cyfra	Pierwsza grupa 6 cyfr	Druga grupa 6 cyfr
0	AAAAAA	CCCCCC
1	AABABB	CCCCCC
2	AABBAB	CCCCCC
3	AABBBA	CCCCCC
4	ABAABB	CCCCCC
5	ABBAAB	CCCCCC
6	ABBBAA	CCCCCC
7	ABABAB	CCCCCC
8	ABABBA	CCCCCC
9	ABBABA	CCCCCC

Kodowanie cyfr:

Cyfra	Kod A	Kod B	Kod 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

### 1.2.3 Obliczanie cyfry kontrolnej kodu kreskowego EAN-13

Suma kontrolna obliczana jest poprzez dodanie iloczynów kolejnych cyfr kodu EAN-13 oraz ich wag. Liczone od końca nieparzyste pozycje mają zawsze wagę 3, a parzyste - wagę 1. Cyfrą kontrolną będzie wartość o jaką należy dopełnić otrzymaną sumę, by ta była podzielna przez 10 np. dla sumy 57 dopełnieniem jest 3, dopełnieniem 42 jest 8 itp.

## 1.3 Cel ćwiczenia

Celem ćwiczenia było stworzenie aplikacji wyświetlającej kody kreskowe EAN-13 stworzone z podanej sekwencji cyfr.

## 2 Realizacja ćwiczenia

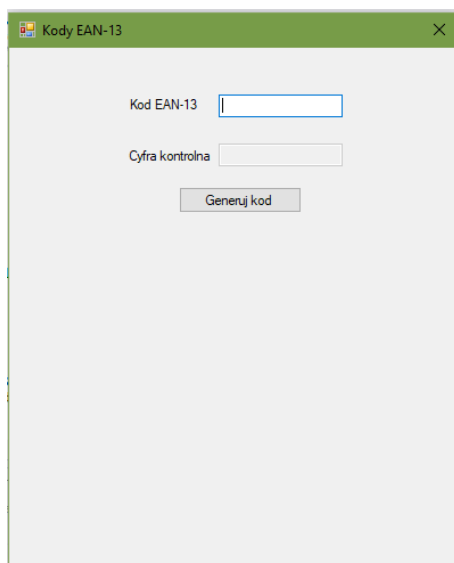
### 2.1 Działanie aplikacji

W ramach ćwiczenia została wykonana aplikacja okienkowa, która generuje kody kreskowe EAN-13. Aplikacja posiada następujące funkcje:

- Generowanie cyfry kontrolnej na podstawie, zadanych przez użytkownika, 12 cyfr kodu EAN-13
- Sprawdzanie poprawności cyfry kontrolnej w przypadku kiedy użytkownik poda 13 cyfr

- Generowanie kodu kreskowego

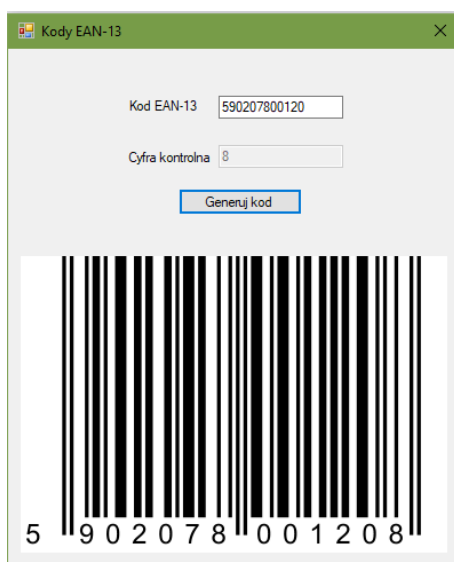
Aplikacja została napisana w języku C#. Graficzny interfejs został przygotowany z użyciem interfejsu Windows Forms platformy .NET Framework.



Rysunek 1: GUI aplikacji po włączeniu

## 2.2 Generowanie kodu kreskowego

W polu tekstowym **Kod EAN-13** należy podać 12 lub 13 cyfrowy kod, następnie nacisnąć **Generuj kod**. W przypadku kiedy użytkownik poda 12 cyfrowy kod, aplikacja samodzielnie obliczy cyfrę kontrolną, wyświetli ją oraz wygenerowany kod kreskowy. Jeśli użytkownik poda 13 cyfrowy kod, program sprawdzi poprawność ostatniej cyfry kontrolnej. Jeśli podana cyfra jest poprawna zostanie wygenerowany kod kreskowy, w przeciwnym wypadku aplikacja poinformuje użytkownika o błędzie występującym w kodzie.



Rysunek 2: GUI aplikacji po wygenerowaniu kodu

## 2.3 Implementacja aplikacji

Aplikacja składa się z trzech klas **Program.cs**, **Form.cs** oraz **Ean.cs**

### 2.3.1 Klasa Program.cs

Klasa Program.cs to główna klasa programu. W metodzie statycznej Main ustawiane są podstawowe parametry aplikacji, oraz uruchamiana jest standardowa pętla aplikacji z wskazaniem na główny formularz klasy Form1.cs, który będzie wyświetlony.

### 2.3.2 Klasa Form.cs

Klasa Windows Forms. Odpowiedzialna jest za GUI aplikacji. Znajdują się w niej wszystkie elementy potrzebne do obsługi formularza aplikacji.

### 2.3.3 Klasa Ean.cs

Klasa Ean zawiera całą logikę aplikacji, jest odpowiedzialna za obliczanie cyfry kontrolnej, zwracanie ciągu znaków (wartości 0 i 1) zakodowanej sekwencji oraz generowanie bitmapy kodu kreskowego.

## 2.4 Najważniejsze funkcje programu

Metoda **CalculateControl()** oblicza i zwraca wartość cyfry kontrolnej na podstawie przechowywanej sekwencji kodu.

```
public int CalculateControl()
{
    int sum = 0;
    for (int i = 1; i < 12; i = i + 2)
        sum += Convert.ToInt32(BarCode.Substring(i, 1));
    sum = sum * 3;
    for (int i = 0; i < 12; i = i + 2)
        sum += Convert.ToInt32(BarCode.Substring(i, 1));
    return (10 - (sum % 10)) % 10;
}
```

Rysunek 3: Funkcja generująca cyfrę kontrolną

Metoda **EncodeCode()** jest odpowiedzialna za kodowanie sekwencji liczbowej. Metoda zwraca ciąg znaków (wartości 0 lub 1) zakodowanej sekwencji 106 modułów (7 ostatnich modułów "Quiet Zone" nie jest wymagane, ponieważ cały kod kreskowy będzie miał białe tło i wymagany rozmiar).

```
private string EncodeCode()
{
    var encoded = "";
    var encoding = _LeftCode[(int)char.GetNumericValue(BarCode[0])];
    // Quiet zone
    encoded += _LeftQuietZone;
    // Marker początku
    encoded += _StartStop;
    // lewa część kodu
    for (var i = 1; i <= 6; i++)
    {
        var digit = (int)char.GetNumericValue(BarCode[i]);
        if (encoding[i - 1] == 'A')
            encoded += _ACode[digit];
        else
            encoded += _BCode[digit];
    }

    // Marker środka
    encoded += _CenterMarker;
    // prawa część kodu
    for (var i = 7; i < 12; i++) encoded += _CCode[(int)char.GetNumericValue(BarCode[i])];
    encoded += _CCode[(int)char.GetNumericValue(CheckSum[0])];

    // Marker końca
    encoded += _StartStop;

    return encoded;
}
```

Rysunek 4: Funkcja odpowiedzialna za kodowanie sekwencji liczbowej

### 3 Podsumowanie i wnioski

Podczas zajęć udało się zrealizować aplikację generującą kody kreskowe EAN13. Poprawność wygenerowanych kodów została potwierdzona za pomocą dostępnego w sali laboratoryjnej skanera kodów kreskowych. Aplikacja ta mogłaby zostać wykorzystana jako część składowa większego systemu pracującego w bibliotece lub sklepie. Podczas tworzenia aplikacji nie zostały napotkane problemy, które uniemożliwiłyby realizację ćwiczenia.

### 4 Źródła

1. <https://pl.wikipedia.org/wiki/EAN>
2. <https://gs1pl.org/standardy/wymiary-kodow-kreskowych-ean-upc/>
3. <https://www.optidata.pl/kody-kreskowe-informacje-czarno-na-bialym/>
4. <https://kody-kreskowe.dlawas.com/ean-13.html>
5. [https://pl.wikipedia.org/wiki/Kod\\_kreskowy](https://pl.wikipedia.org/wiki/Kod_kreskowy)