

```
Public Class Form1
```

```
'Program kreslil: oklufc  
' a vidlácky dopsal následující řádky:
```

```
Dim B7, B6, B5, B4, B3, B2, B1, B0 As String 'Deklarace znaků v řetězu bitového  
zobrazení  
Dim N1 As Byte 'pomocná proměnná s hodnotami 0 až 255
```

```
'Následující dvě procedury jsou supervidlácké. Vycházejí z principu, že komponenty  
'NumericUpDown umí hodnotu (Value) zobrazovat jako hexadecimální číslo nebo dekadické  
číslo.
```

```
'Aniž bychom tedy přemýšleli, čím je ve skutečnosti hodnota reprezentována, zobrazíme  
ji jednou komponentou jako hex a
```

```
'jinou jako dekadické.
```

```
'Na to stačí dva řádky, po jednom v proceduře obsluhy události o změně u každé  
komponenty. Jednoduché, že?
```

```
'V proceduře Bity() se řeší binární zobrazení jednotlivých bitů jako nul a jedniček.
```

```
'Zobrazuje se to s nejvýznamnějším bitem vpravo i vlevo, jak je kdo zvyklý.
```

```
'Užitečnost takové kalkulačky je u systémů, kde se pomocí hodnoty bajtu
```

```
'nastavují např. úroveň L a H na portech čipu.
```

```
'Pokud se vám 0 a 1 nelíbí, přepište si to sami na L a H
```

```
'Hlavně to nepřepisujte na True a False, to dělají "poormen", kteří neviděli sondu  
nebo voltmetr.
```

```
Private Sub NumericUpDown1_ValueChanged(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles H.ValueChanged
```

```
    N.Value = H.Value 'když se změní hex hodnota, změní se i desítková hodnota
```

```
    Bity() 'skok do procedury, která vypočte a zobrazí hodnoty bitů
```

```
End Sub
```

```
Private Sub NumericUpDown2_ValueChanged(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles N.ValueChanged
```

```
    H.Value = N.Value 'když se změní desítková hodnota, změní se i hex hodnota
```

```
End Sub
```

```
Private Sub Bity() 'procedura, která vypočte, seřadí a zobrazí hodnoty  
jednotlivých bitů jako nuly a jedničky
```

```
    B7 = "0" 'před výpočtem všechny bity na 0; B7 je nejvýznamnější
```

```
    B6 = "0"
```

```
    B5 = "0"
```

```
    B4 = "0"
```

```
    B3 = "0"
```

```
    B2 = "0"
```

```
    B1 = "0"
```

```
    B0 = "0"
```

```
    N1 = N.Value 'N1 nabere hodnotu z N.Value
```

```
    If (N1 / 128) >= 1 Then 'N1 se testuje na nejvyšší bit tak, že se vydělí jeho  
hodnotou (/128)
```

```
        N1 = N1 - 128 'N1 nabývá zbytek po dělení, tj. odečte se hodnota  
nejvyššího řádu
```

```
        B7 = "1" 'bit B7 se nastavuje na 1
```

```
    End If
```

```
    If (N1 / 64) >= 1 Then 'testování na nižší bit (řád) dvojkové soustavy,  
obdoba předchozích řádků
```

```
        N1 = N1 - 64
```

```
        B6 = "1"
```

```
    End If
```

```
    If (N1 / 32) >= 1 Then
```

```
        N1 = N1 - 32
```

```
        B5 = "1"
```

```
    End If
```

```
    If (N1 / 16) >= 1 Then
```

```
        N1 = N1 - 16
```

```
        B4 = "1"
```

```
    End If
```

```
If (N1 / 8) >= 1 Then
    N1 = N1 - 8
    B3 = "1"
End If

If (N1 / 4) >= 1 Then
    N1 = N1 - 4
    B2 = "1"
End If

If (N1 / 2) >= 1 Then
    N1 = N1 - 2
    B1 = "1"
End If

If N1 >= 1 Then
    'testování na nejnižší bit (řád) dvojkové soustavy
    'tento zbytek už k ničemu nepotřebujeme
    B0 = "1"
    'nastavení nejméně významného bitu B0, tj. LSB
End If

LSB.Text = B0 + B1 + B2 + B3 + " " + B4 + B5 + B6 + B7
vlevo
MSB.Text = B7 + B6 + B5 + B4 + " " + B3 + B2 + B1 + B0
vlevo
'zobrazení bitů, LSB je
'zobrazení bitů, MSB je

End Sub

End Class
```