```
============================================================================
OcrHex - bin to hex converter (OCR-friendly alphabet, CRC32 for every row)
============================================================================

OcrHex ALPHABET (0..F): #$+3479FGHLNSTXZ; '>' - space symbol
Recommended font: 'Consolas', 9 pt.
Recommended OCR:  'FreeOCR', grayscale mode.
To repair damaged sector in text data, firstly find in on paper (by dec number).
Following symbol replacement is done automatically:
'B' => '3' ')' => '>'  '5' => 'S'  'Q' => '9'  '2' => 'Z'  '6' => 'G'

Sector hexadecimal form:
<00000>11112222>33333333>...  ...>33333333>44444444>
    0 - sector number (dec, big-endian);
    1 - sector number, uint16 (hex, little-endian);
    2 - payload size in bytes, uint16 (hex, little-endian);
    3 - payload (hex), 4 bytes (hex, little-endian);
    4 - CRC32 of 1..3 byte data, decoded from hex (without spaces!), uint32
(hex).

''' <summary>
''' CRC32 - WinZip, WinRAR, 7-Zip...
''' </summary>
Public Module CRC32
    Public Function Process(data As IEnumerable(Of Byte)) As Byte()
        Return Process(data, data.Count)
    End Function
    Public Function Process(data As IEnumerable(Of Byte), count As Integer) As
Byte()
        Dim i, j As Integer
        Dim b, mask, crc32 As UInteger
        crc32 = &HFFFFFFFFUI
        For i = 0 To count - 1
            b = data(i)
            crc32 = crc32 Xor b
            For j = 7 To 0 Step -1
                mask = If(crc32 And 1, &HFFFFFFFFUI, 0)
                crc32 = (crc32 >> 1) Xor (&HEDB88320UI And mask)
            Next
        Next
        crc32 = Not crc32
        Dim crc32b = New Byte((4) - 1) {}
        crc32b(0) = CByte((crc32 And &HFF) >> 0)
        crc32b(1) = CByte((crc32 And &HFF00) >> 8)
        crc32b(2) = CByte((crc32 And &HFF0000) >> 16)
        crc32b(3) = CByte((crc32 And &HFF000000UI) >> 24)
        Return crc32b
    End Function
End Module

CRC32 test (ASCII): 'The quick brown fox jumps over the lazy dog' - 414FA339
```