Bit String Pattern

There are 2^n bit strings of length *'n'*.
Bit values are: 0 or 1

Example A: bit string of length 1, n = 1

$2^1 = 2$, so there will be 2 unique bit strings of values –

Bit string #	Bit String Represented	Decimal (base 10) Equivalent Value
1	0	0
2	1	1

Example B: bit string of length 4, n = 4

$2^4 = 16$, so there will be 16 unique bit strings of values –

Bit string #	Bit String Represented				Decimal (base 10) Equivalent Value	Hexadecimal (base 16) Equivalent Value
	$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$		
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						

Converting from Binary to Hexadecimal or Hexadecimal to Binary:
*It takes a bit string of **length 4** to represent each hexit. Use table above to convert values, always grouping bit strings from **RIGHT to LEFT**. (Note: you must use 4 bits when completing the conversion, but you may remove leftmost/padded zeros upon completion of the conversion.)*

$10\ \ 1011\ \ 0101_2$

$2\quad B\quad 5_h$

$4\quad E\quad 2\quad 1_h$

$0100\ \ 1110\ \ 0010\ \ 0001_2 = 100111000100001_2$

Converting from Binary to Octal or Octal to Binary:
*It takes a bit string of **length 3** to represent each octit. Use table above (yellow shaded region) to convert values, always grouping bit strings **from RIGHT to LEFT**. (Note: you must use 3 bits when completing the conversion, but you may remove leftmost/padded zeros upon completion of the conversion.)*

$1\ \ 010\ \ 110\ \ 101_2$

$1\quad 2\quad 6\quad 5_8$

$7\quad 3\quad 4\quad 1_8$

$111\ \ 011\ \ 100\ \ 001_2$