ELETRÔNICA DIGITAL – SISTEMAS NUMÉRICOS

Distinção entre o sistema digital e analógico:

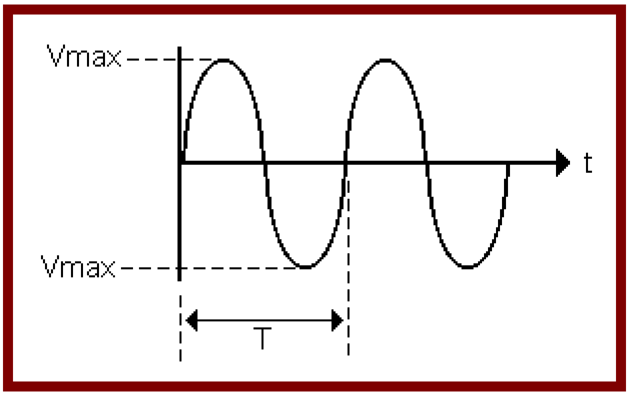
As técnicas e os sinais eletrônicos são divididos em dois ramos:

1. sinais analógicos
2. sinais digitais

Sinal analógico:

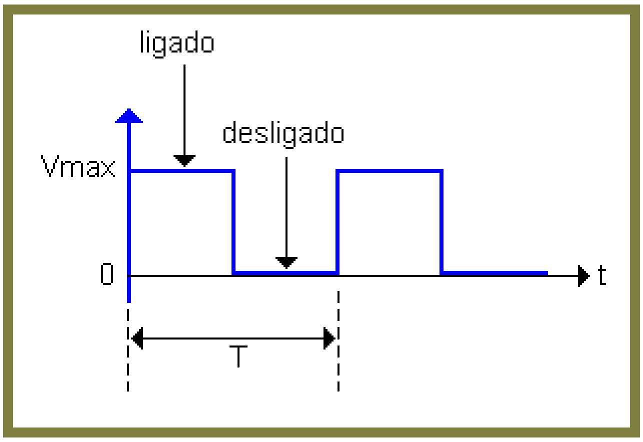
O sinal analógico varia de forma contínua em função do tempo, como por exemplo, uma tensão alternada, representada por uma onda senoidal, cuja variação é sempre contínua, variando do zero até um valor máximo positivo, retornando ao zero e variando a partir daí a um valor máximo negativo, retornando novamente ao zero.

Tudo isso ocorre durante um período, o qual denominamos “T”. Veja a figura a seguir:



Sinal digital:

O sinal digital varia através de passos distintos em função do tempo, como por exemplo, a saída de um manipulador de Código Morse, ou qualquer outro sistema que possui apenas duas condições distintas: LIGADO e DESLIGADO.



RESUMINDO:

Quando nos referimos a um sinal analógico, podemos comparar com a tensão elétrica residencial, cuja frequência é de 60Hz. Isto significa que a variação dessa forma de onda é sempre contínua na unidade de tempo.

Quando nos referimos a um sinal digital, podemos estabelecer uma analogia a um determinado ambiente com uma lâmpada e um interruptor.

Acionando o interruptor podemos determinar as duas condições do ambiente: “iluminado” ou “não iluminado”. Isto significa que o interruptor apresenta apenas duas condições: *ligado ou desligado*.

Veremos mais adiante, que na lógica binária a condição “ligado” é referida como nível lógico 1 e a condição “desligado” como nível lógico 0.

SISTEMAS NÚMERICOS

Sistema digital:

O sistema numérico que o ser humano entende é o “sistema decimal”, que é composto de 10 dígitos que compreendem 0 a 9. Por isso é o sistema mais usado.

Esse sistema é chamado de sistema “base 10”, baseando-se nas potências de 10, podendo dessa forma representar qualquer número ou quantidade.

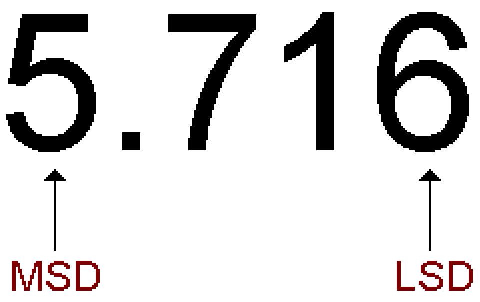
Em Eletrônica Digital vamos estudar os sistemas de base 2, base 8 e base 16 respectivamente: sistema binário, sistema octal e sistema hexadecimal.

No entanto, para entender esses sistemas é necessário entender como podemos manipular o sistema decimal, que é o sistema nativo da linguagem humana.

Tomemos como exemplo o número decimal 5.716.

O dígito 5 é o mais significativo (MSD) e o dígito 6 o menos significativo (LSD).

MSD = Most Significant Digit
LSD = Least Significant Digit



A maneira mais comum de representar um número decimal é multiplicar cada dígito ou algarismo pela base, elevando-a a uma determinada potência. No sistema decimal, como já vimos é base 10.

Basta aplicar uma fórmula bem simples, associando-a ao MSD:

$$X^{n-1}$$

Onde:

X é a base do sistema
n é a quantidade de dígitos do número

Teremos então associado ao MSD: 10^{4-1} ➔ 10^3

Na medida em que os dígitos forem se tornando menos significativos, o índice da potência de 10 vai diminuindo, até terminar em 10^0.

Veja como fica a decomposição do número 5.716

5 x 10^3 = 5 x 1.000 = **5.000**
7 x 10^2 = 7 x 100 = **700**
1 x 10^1 = 1 x 10 = **10**
6 x 10^0 = 6 x 1 = **6**

Somando 5.000 + 700 + 10 + 6, teremos 5.716

OBS: Qualquer número elevado à potência "0" é sempre igual a "1"

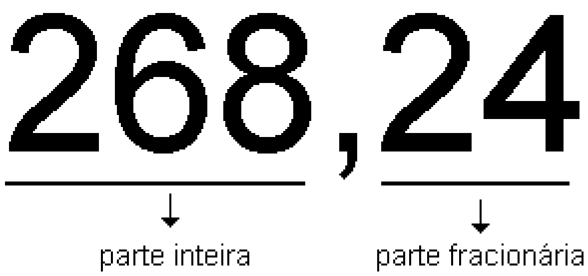
Podemos também ao decompor o número 5.716, associar o LSD à potência de 10 elevada a "zero" (10^0).

Neste caso, na medida em que os dígitos forem se tornando mais significativos, o índice da potência de 10 vai aumentando, porém, não deve ultrapassar 10^{n-1}.

Decompondo um número decimal fracionário:

Tomemos como exemplo o número decimal 268,24.

268 é a parte inteira do número e 24 é a parte decimal desse número.



A novidade é que para a parte fracionária, ao primeiro dígito à direita da vírgula devemos associar em potência de 10 com índice negativo, começando a partir de -1 e incrementando sucessivamente até a quantidade dos dígitos fracionários.

Então, a parte inteira ficará assim:

2 x 10^2 = 2 x 100 = **200**
6 x 10^1 = 6 x 10 = **60**
8 x 10^0 = 8 x 1 = **8**

Decompondo a parte fracionária:

$\mathbf{2} \times 10^{-1} = 2 \times 1/10 = 2 \times 0,1 = \mathbf{0,2}$
$\mathbf{4} \times 10^{-2} = 2 \times 1/100 = 4 \times 0,01 = \mathbf{0,04}$

Somando: 200 + 60 + 8 + 0,2 + 0,04, teremos: **268,24**

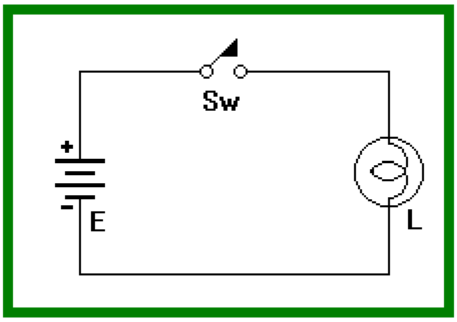
Sistema binário:

Em virtude de sua extraordinária simplicidade, o sistema binário é muito utilizado no processamento de informações em computadores, equipamentos digitais de transmissão e recepção, vídeo games, seletores de canais de TV, enfim, inúmeras aplicações.

É um sistema que possui apenas 2 estados, 0 ou 1.

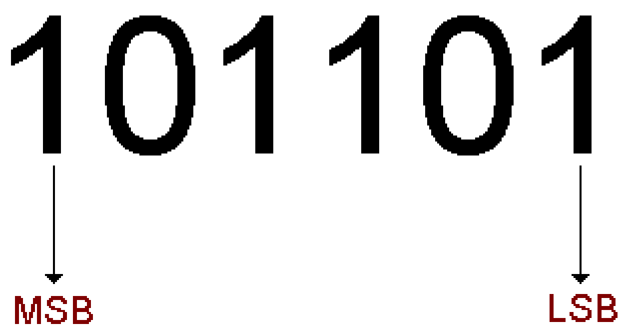
Enquanto que no sistema decimal temos que representar 10 estados (0 a 9) no sistema binário é muito mais simples, pois esses estados podem ser representados por qualquer componente ou circuito eletrônico do tipo “liga-desliga”.

Se considerarmos um circuito eletrônico bem simples, composto de uma chave (interruptor) e uma lâmpada, conforme ilustra a figura abaixo, teremos;



Chave aberta = NL 0	Lâmpada apagada = NL 0
Chave fechada = NL 1	Lâmpada acesa = NL 1

O sistema binário deve ter muito bem caracterizado o seu dígito mais significativo e o menos significativo, que passaremos a denominar “bit”. Assim teremos: LSB (Least Significant Bit) e MSB (Most Significant Bit).



No entanto, o ser humano não entende a linguagem binária, a não ser que a mesma seja convertida em linguagem decimal.

Da mesma forma um computador não entende a linguagem decimal, ou seja, só opera com linguagem binária. Assim, é preciso fazer as conversões *binário-decimal* e *decimal-binário* para que ocorra um processo operacional.

Este processo denomina-se conversão entre bases.

Conversão do sistema binário para o sistema decimal (base 2 para base 10)

A menos que no enunciado seja especificada a base de um sistema, o mesmo deve ser identificado por um subíndice e isto vale para qualquer base.

Exemplo 1: Converter para base 10 o binário 110101

O binário 110101 possui 6 dígitos (6 bits). O bit da estrema esquerda é o mais significativo (MSB) e o da extrema direita o menos significativo (LSB), como acontece no sistema de numeração decimal.

Devemos então associar ao MSB ➔ X^{n-1}

X = 2 (base do sistema)
n = 6 (quantidade de bits)

teremos então: 2^5

$\mathbf{1} \times 2^5 = 1 \times 32 = \mathbf{32}$
$\mathbf{1} \times 2^4 = 1 \times 16 = \mathbf{16}$
$\mathbf{0} \times 2^3 = 0 \times 8 = \mathbf{0}$
$\mathbf{1} \times 2^2 = 1 \times 4 = \mathbf{4}$
$\mathbf{0} \times 2^1 = 0 \times 2 = \mathbf{0}$
$\mathbf{1} \times 2^0 = 1 \times 1 = \mathbf{1}$

Somando: 32 + 16 + 4 + 1 = 53

$110101_2 = 53_{10}$

Podemos também associar ao LSB a base do sistema elevada a zero (2^0).

Para facilitar a conversão, apresentamos a seguir uma tabela de potência de 2.

$2^0 = 1$	$2^{11} = 2.048$
$2^1 = 2$	$2^{12} = 4.096$
$2^2 = 4$	$2^{13} = 8.192$
$2^3 = 8$	$2^{14} = 16.384$
$2^4 = 16$	$2^{15} = 32.768$
$2^5 = 32$	$2^{16} = 65.536$
$2^6 = 64$	$2^{17} = 131.072$
$2^7 = 128$	$2^{18} = 262.144$
$2^8 = 256$	$2^{19} = 524.288$
$2^9 = 512$	$2^{20} = 1.048.576$
$2^{10} = 1.024$	

Observe que à medida que a potência é incrementada em 1, o resultado dobra.

Bit, nibble e byte

Bit = a menor unidade de informação binária

Nibble = conjunto de 4 bits

Byte = conjunto de 8 bits

Na tabela mostrada acima, 2^{10} = corresponde a 1 kilobit (quilobit) e 2^{20} corresponde a 1 megabit.

Exemplo 2: Converter para base 10 o binário 100000001

O binário em questão possui 9 bits. Logo, ao MSB associa-se $2^{9-1} = 2^8$
Ao LSB associa-se então $2^0 = 1$

Consultando a tabela, $2^8 = 256$

Logo o resultado da conversão corresponde ao decimal **257** (256 + 1)

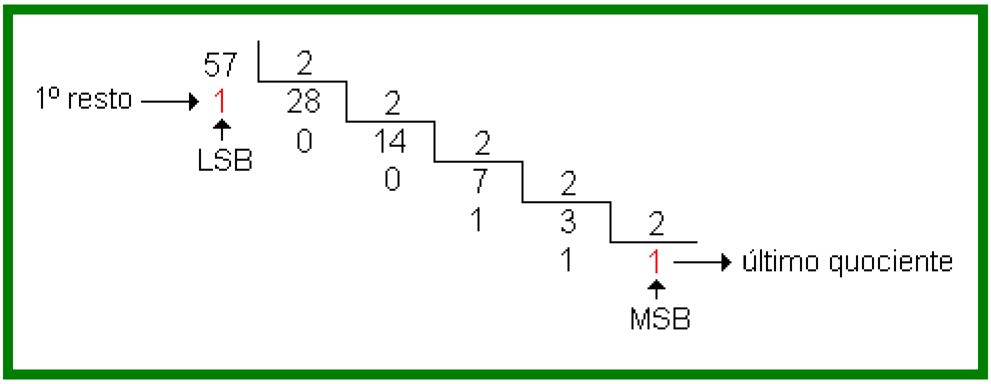
Conversão do sistema decimal para o sistema binário (base 10 para base 2)

Consideremos como exemplo o número decimal 57.

Para a conversão para a base 2, o mesmo deve ser dividido sucessivamente por 2, de tal forma que o resto de cada divisão não deve ser maior do que 1 (o resto só pode assumir valores 0 ou 1).

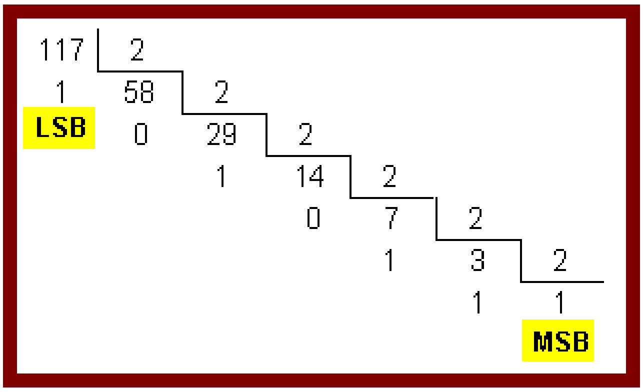
A leitura é feita a partir do último quociente que é o bit mais significativo (MSB) e o bit menos significativo (LSB) é o primeiro resto.

Veja a seguir o diagrama do processo de divisão, onde o primeiro resto é o LSB e o último quociente é o MSB.



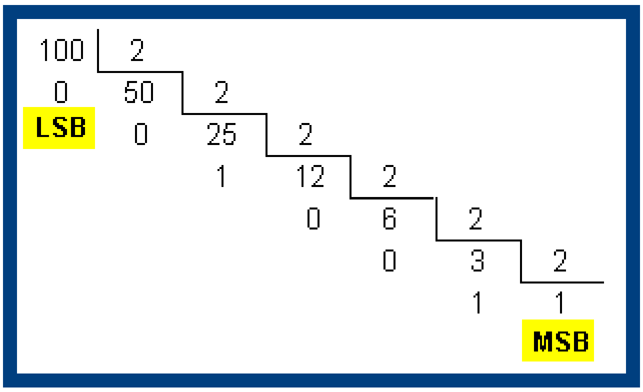
Assim, 57_{10} equivale a 111001_2

Exemplo 1: Converter para base 2 o decimal 117



$117_{10} = 1110101_2$

Exemplo 2: Converter para base 2 o decimal 100



$100_{10} = 1100100_2$

Números binários fracionários:

A exemplo de um número decimal, a decomposição de um número binário fracionário é feita de forma idêntica, ou seja, acrescenta-se à direita da vírgula potência negativa de 10, a partir do índice -1.

Supondo: o número binário 11,11

Convertendo para decimal, temos:

11,11

parte inteira — parte fracionária

Parte inteira:

1 x 2^1 = 1 x 2 = **2**
1 x 2^0 = 1 x 1 = **1**

Parte fracionária:

1 x 2^{-1} = 1/2 = **0,5**
1 x 2^{-2} = 1/4 = **0,25**

Parte inteira: 2 + 1 = 3
Parte fracionária: 0,5 + 0,25 = 0,75

$11,11_2 = 3,75_{10}$

Exemplo: Converter para decimal o binário 10111,011

A parte inteira do binário possui 5 dígitos, logo, associaremos ao MSB $2^{n-1} = 2^4$

A parte fracionária possui 3 bits, logo, associaremos imediatamente após a vírgula 2^{-1}, 2^{-2} e 2^{-3}

Convertendo:

1 x 2^4 = 1 x 16 = **16**
0 x 2^3 = 0 x 8 = **0**
1 x 2^2 = 1 x 4 = **4**
1 x 2^1 = 1 x 2 = **2**
1 x 2^0 = 1 x 1 = **1**

0 x 2^{-1} = **0**
1 x 2^{-2} = 1/4= **0,25**
1 x 2^{-3} = 1/8 = **0,125**

Parte inteira = **23**; parte fracionária = **0,375**

Assim: $10111,011_2 = 23,375_{10}$

OBS: para resolver um número com expoente negativo, basta invertê-lo e trocar a polaridade do expoente. Por exemplo:

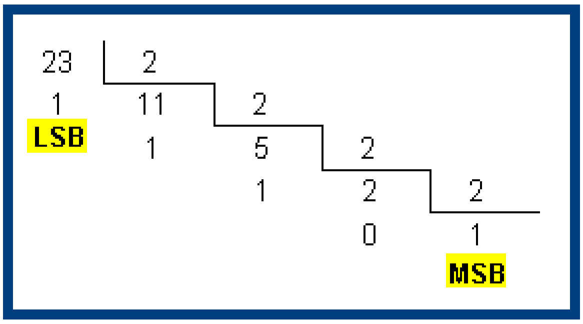
$$2^{-3} = 1/2^3 = 1/8 = 0,125$$

Veja a seguir uma tabela com expoentes negativos de -1 até -10, na base 2.

2^{-1}	0,5
2^{-2}	0,25
2^{-3}	0,125
2^{-4}	0,0625
2^{-5}	0,03125
2^{-6}	0,015625
2^{-7}	0,0078125
2^{-8}	0,00390625
2^{-9}	0,001953125
2^{-10}	0,0009765625

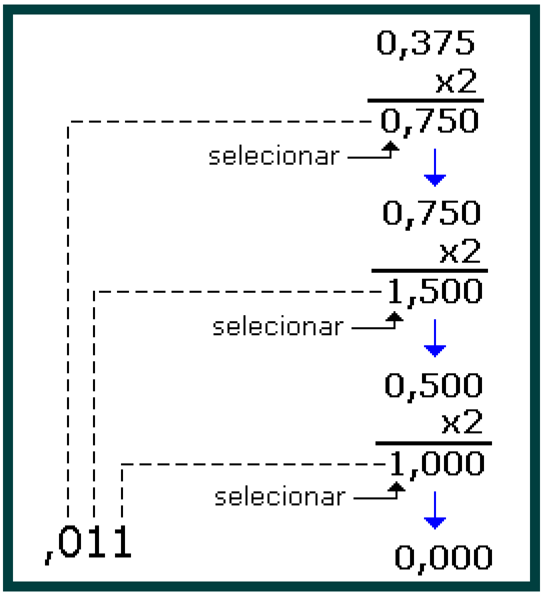
Conversão do decimal fracionário para o binário:

Supondo o decimal 23,375. Para convertê-lo para binário, devemos primeiramente converter a parte inteira, no caso, 23.



$23^{10} = 10111_2$

Convertendo a parte fracionária 0,375:

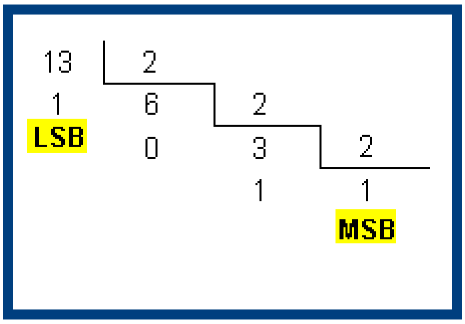


$0,375_{10} = 0,011_2$

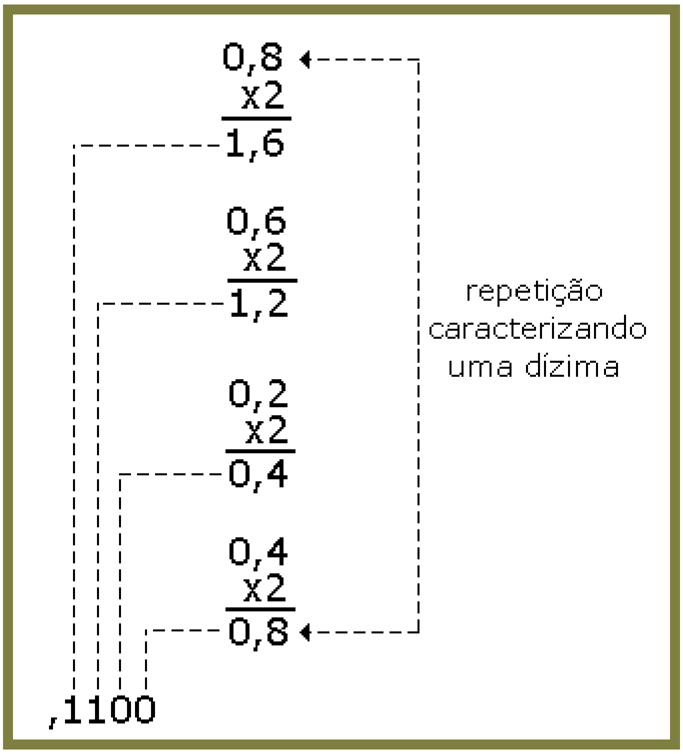
Portanto: $23,375_{10} = 10111,011_2$

Exemplo: Converter em binário o número decimal 13,8

Convertendo a parte inteira do número:



Convertendo a parte fracionária



Portanto: $13,8_{10} = 1101,1100..._{2}$

SISTEMA OCTAL

O sistema octal possui 8 dígitos, de 0 a 7. Assim, qualquer número cuja base é 8, pode ser expresso em potência de 8.

O processo para a conversão de bases é idêntico ao que foi visto anteriormente, exceto que neste caso utilizaremos potências de 8, por ser sistema de base 8 (octal).

Tomemos como exemplo o número 231 na base octal (231_8)

Para converter para base 10, basta associar ao MSB, X^{n-1} neste caso, 8^2.

Convertendo:

$$\mathbf{2} \times 8^2 = 2 \times 64 = \mathbf{128}$$
$$\mathbf{3} \times 8^1 = 3 \times 8 = \mathbf{24}$$
$$\mathbf{1} \times 8^0 = 1 \times 1 = \mathbf{1}$$

Portanto: $231_8 = 153_{10}$

Para facilitar o cálculo das conversões, podemos construir uma tabela com potências de 8, conforme mostrado abaixo de 8^1 até 8^8, lembrando que qualquer número elevado à potência 0 é igual a 1.

$$\mathbf{8^1 = 8}$$
$$\mathbf{8^2 = 64}$$
$$\mathbf{8^3 = 512}$$
$$\mathbf{8^4 = 4.096}$$
$$\mathbf{8^5 = 32.768}$$
$$\mathbf{8^6 = 262.144}$$
$$\mathbf{8^7 = 2.097.152}$$
$$\mathbf{8^8 = 16.777.216}$$

Exemplo: Converter para o sistema decimal o número na base octal 1.130

Tendo o número 4 dígitos, então associaremos ao MSB, 8^3.

$$\mathbf{1} \times 8^3 = 1 \times 512 = \mathbf{512}$$
$$\mathbf{1} \times 8^2 = 1 \times 64 = \mathbf{64}$$
$$\mathbf{3} \times 8^1 = 3 \times 8 = \mathbf{24}$$
$$\mathbf{0} \times 8^0 = 0 \times 1 = \mathbf{0}$$

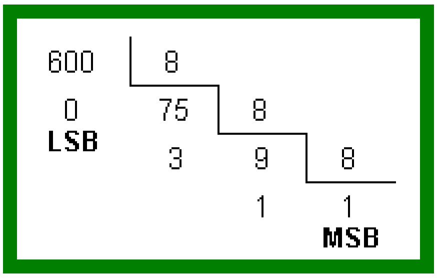
Portanto: $1.130_8 = 600_{10}$

Fazendo a conversão do sistema decimal para o sistema octal:

Seguindo os métodos anteriores, basta dividir o número decimal sucessivamente por 8, que é o base do sistema.

Os restos e o último quociente não podem ser maiores do que 7.

Vamos então inverter o processo de conversão do último exercício, convertendo 600 base decimal (600_{10}) para base octal.



$600_{10} = 1.130_8$

SISTEMA HEXADECIMAL

É atualmente o sistema mais utilizado em computação. É formado por 16 dígitos alfanuméricos, conhecido como sistema de base 16. Veja a equivalência entre o sistema decimal e hexadecimal:

DECIMAL	HEXADECIMAL
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F

O processo de conversão é análogo ao visto anteriormente, porém a base é 16.

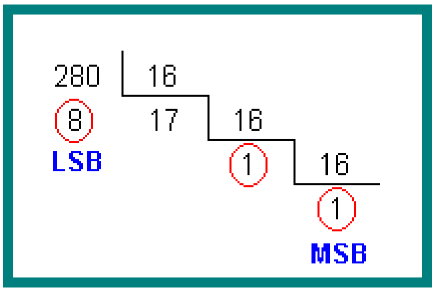
Por exemplo: Converter para o sistema decimal o número hexadecimal 118.

O número possui 3 dígitos, logo, associaremos ao MSB: $X^{n-1} = 16^2$

$$\mathbf{1} \times 16^2 = 1 \times 256 = \mathbf{256}$$
$$\mathbf{1} \times 16^1 = 1 \times 16 = \mathbf{16}$$
$$\mathbf{8} \times 16^0 = 8 \times 1 = \mathbf{8}$$

$$118_{16} = 280_{10}$$

Invertendo o processo, ou seja, convertendo 280 decimal para base 16, basta dividir sucessivamente por 16.



O primeiro resto é o LSB ou LSD e o último quociente é o MSB. Ambos não podem ser maiores do que 15.

$$\text{Portanto: } 280_{10} = 118_{16}$$

$16^1 = 16$
$16^2 = 256$
$16^3 = 4.096$
$16^4 = 65.536$
$16^5 = 1.048.576$
$16^6 = 16.777.216$
$16^7 = 268.435.456$
$16^8 = 4.294.967.296$

Exemplo: Converter para o sistema decimal o número hexadecimal 2FC7

No sistema hexadecimal temos:

F = 15
C = 12

2 (15)(12)7
F C

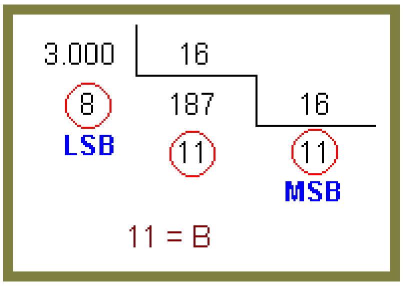
O número possui 4 bits ou dígitos, logo, associaremos ao MSB: 16^3

Convertendo:

2 x 16^3 = 2 x 4.096 = **8.192**
15 x 16^2 = 15 x 256 = **3.840**
12 x 16^1 = 12 x 16 = **192**
7 x 16^0 = 7 x 1 = **7**

Portanto: $2FC7_{16} = 12.231_{10}$

Converter o número decimal 3.000 para base hexadecimal



$3.000_{10} = BB8_{16}$