Introducción

Dos sistemas de numeración de especial importancia para nosotros por su aplicación en la computación son el octal y el hexadecimal.

Vamos a ver como podemos a partir de una representación obtener otra. Estamos familiarizados con el sistema decimal, pero cualquier numero en este sistema podemos verlo de otra forma al representarlo en otro sistema. Es importante no perder de vista que estamos representando el mismo numero de distintas bases, según el sistema de numeración elegido, pero sigue teniendo el mismo valor.

Octal

El sistema octal es un sistema de numeración posicional de base 8. Los símbolos que se usan en este sistema son:

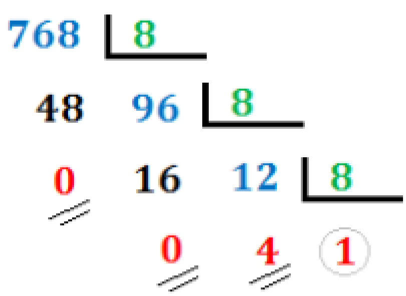
##0, 1, 2, 3, 4, 5, 6, 7##

Para indicar que un número está escrito en base 8, usamos el subíndice $_{(8)}$, y para indicar que un número está escrito en base 10, usamos el subíndice $_{(10)}$.

$10_{(8)}=8_{(10)}$
$21_{(8)}=17_{(10)}$
$102_{(8)}=66_{(10)}$

A continuación, explicamos el método para pasar del sistema decimal al sistema octal mediante un ejemplo. Pasamos el número $768_{(10)}$ a base 8:

- Dividimos el número por 8
- Si el cociente es >= 8, lo volvemos a dividir por 8
- Continuamos hasta obtener un cociente menor a 8

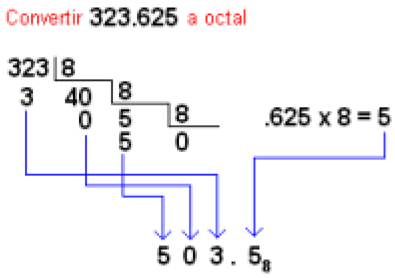


El número obtenido es ##1400## $_{(8)}$

El cociente es 1, menor que 8, con lo que hemos terminado el proceso. Hemos indicado los restos con dos rayas y el último cociente con una circunferencia.

Pasaje de números con coma en base 10 a base 8:

Para números con coma, trabajamos la parte decimal de forma similar al proceso que aplicamos para pasar a binario, tomamos solo la parte decimal y se multiplica por 8, por ser la base a la que se quiere llevar. Si en la multiplicación que se realiza, da una parte entera, esa se deja sera el dígito que se usara en el octal, y con la parte decimal restante se repite el procedimiento.



Pasaje de un número Octal a uno decimal:

Lo que se realiza en este pasaje es tener en cuenta el peso de cada posición en una cifra octal. Por ejemplo para convertir el número $237_{(8)}$ a decimal se hace el valor de cada dígito por la base elevada a la posición donde se encuentra el mismo.

$2* (8^2)+ 3* (8^1) + 7* (8^0) = 2*64 + 3*8 + 7*1 = 159_{(10)}$

$273_{(8)} = 159_{(10)}$

Hexadecimal

El sistema hexadecimal es un sistema de numeración posicional de base 16. Esto quiere decir que necesitamos 16 símbolos. Vamos a obtener estos 16 símbolos usando letras para los valores de 10 en adelante, es decir

Para representar **10** usamos el símbolo **A**
Para representar **11** usamos el símbolo **B**
Para representar **12** usamos el símbolo **C**
Para representar **13** usamos el símbolo **D**
Para representar **14** usamos el símbolo **E**
Para representar **15** usamos el símbolo **F**

Entonces, los símbolos que se usan en este sistema son:

##0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F##

Las equivalencias de los primeros 16 números del sistema decimal se ve en la siguiente tabla:

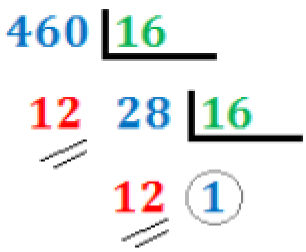
Decimal	Hexadecimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7

Decimal	Hexadecimal
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F

Para indicar que un número está escrito en base 16, usamos el subíndice (16) o el subíndice (hex), y para indicar que un número está escrito en base 10, usamos el subíndice (10).

Pasaje de Base 10 a 16:

Utilizamos el método de divisiones sucesivas por la base 16 hasta que no podamos dividir más el cociente que nos va quedando.



Nos queda el número ##1CC## $_{(16)}$

Pasaje de base 16 a Base 10:

Lo que se realiza en este pasaje es tener en cuenta el peso de cada posición en una cifra Hexadecimal. Por ejemplo para convertir el número A37F Base 16 a Base decimal se hace el valor de cada dígito por la base elevada a la posición donde se encuentra el mismo.

- $A37F_{16} = 10 \times 16^{3} + 3 \times 16^{2} + 7 \times 16^{1} + 15 \times 16^{0} = 41855_{\ (10)}$

Notemos que a la hora de resolver el calculo, reemplazamos la letra por el valor que representa.

Reflexión sobre los números binarios

Vamos a analizar cierta característica de 8 y 16, las bases en las que estamos trabajando.

8 es una potencia de 2, en particular $\mathbf{2^{3}}$ Si los escribimos en binario seria $\mathbf{1000_{(2)}}$ Recordemos los primeros números en binario

Binario	Decimal
000	0
001	1
010	2
011	3

Binario	Decimal
100	4
101	5
110	6
111	7

Como vemos en **3** lugares se pueden representar todos los valores que representan los símbolos que se usan en un dígito octal.

Entonces podemos decir que **2^3-1**, es el máximo valor representable en **3** posiciones, recordar que **2^3** es **8**

Vamos ahora a ver que pasa con base 16

16 también es una potencia de **2**, en particular **2^4** Si los escribimos en binario seria **$10000_{(2)}$** Recordemos los primeros números en binario

Binario	Decimal	Hexadecimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	10	A
1011	11	B
1100	12	C
1101	13	D
1110	14	E
1111	15	F

En este caso usamos **4** lugares y también se pueden representar todos los valores que representan los símbolos que se usan en un dígito hexadecimal.

Entonces podemos decir que **2^4-1**, es el máximo valor representable en **4** posiciones, recordar que **2^4** es **16**

Esta es una característica que se repetirá siempre

Si **N = 2^n**, entonces **2^n-1**, es **N-1**, que sera el mayor valor representable en **n** posiciones, y el rango de valores representados en **n** dígitos va de **0 a N-1**, son **N** valores distintos

Con números concretos seria Si N=16 entonces es n=4

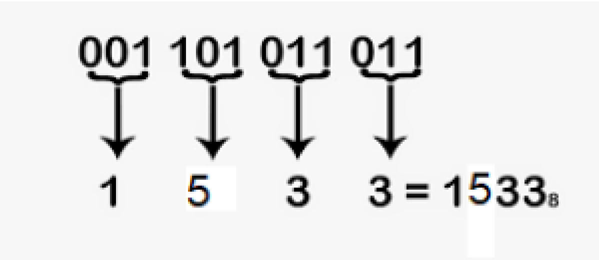
16 = 2^4 En 4 posiciones el numero mas grande a representar es 15 o sea N-1 El total de valores representados en 4 dígitos es 16 (de 0 a 15)

Pasaje de un número Binario a Octal

Como vimos en el punto anterior en 3 posiciones se pueden representar todos los símbolos utilizados en base 8, vamos a aprovechar esta característica para realizar el pasaje de binario a octal.

Si tomamos un numero en binario y agrupamos sus dígitos de a tres, cada grupo de tres representara un dígito octal

Ejemplo:



Tomamos cada grupo de tres y calculamos el valor que representa, cada grupo de 3 se transforma en un dígito del numero octal.

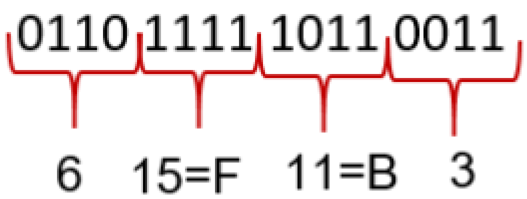
Comprobación

Si ahora tomamos el numero obtenido $1733_{(8)}$ y lo llevamos a base 10 y por otro lado tomamos el binario $001111011011_{(2)}$ y lo llevamos a base 10, deberían dar el mismo valor.

¿Por que podemos asegurar esto?

Pasaje de Binario a Hexadecimal

Nuevamente nos encontramos en una base que es potencia de 2 ($16=2^4$), entonces podemos construir en 4 posiciones todos los símbolos necesarios para construir un numero hexadecimal

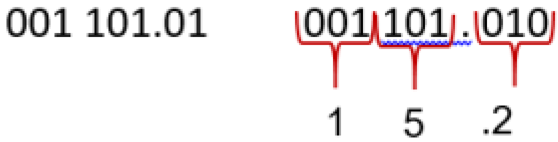


Obtenemos el numero ##6FB3## $_{(16)}$

Tanto en la conversión a Octal como a Hexadecimal, podemos completar con ceros a la izquierda que como sabemos no le agregan valor al numero, para de esta forma tener un estándar en el agrupamiento de bits (dígitos)

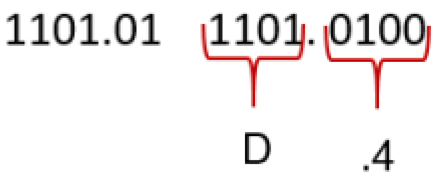
Y que pasa cuando tenemos números binarios con decimales? Tanto para pasaje a Octal como a Hexadecimal, vamos a completar también los dígitos decimales con cero pero teniendo cuidado de no afectar el valor del numero entonces en este caso se debe completar a la derecha

En un pasaje a octal



Completamos dos 0 a la izquierda y uno a la derecha, después de la coma. Agrupamos de a 3 bits y nos queda ##15.2##$_{(8)}$

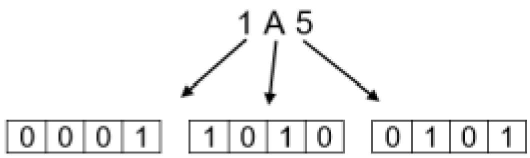
Y en el caso de Hexadecimal



Completamos los grupos de 4 bits, tanto para la parte entera como la parte decimal, pero cuidando de no afectar el valor del numero. Agrupamos de a cuatro, obtenemos ##D.4##$_{(16)}$

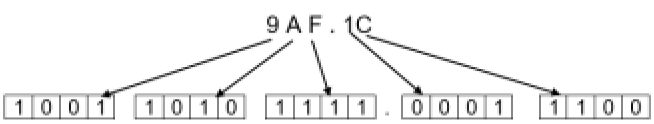
Pasaje de Hexadecimal a Binario:

Para realizar el pasaje de un número Hexadecimal a Binario, lo que hacemos es representar cada dígito hexadecimal en 4 bits en binario, estamos haciendo la inversa de lo que hacemos para pasar de binario a hexadecimal. Debemos tener presente que valor representan los símbolos A,B,C,D,E,F. Veamos un ejemplo



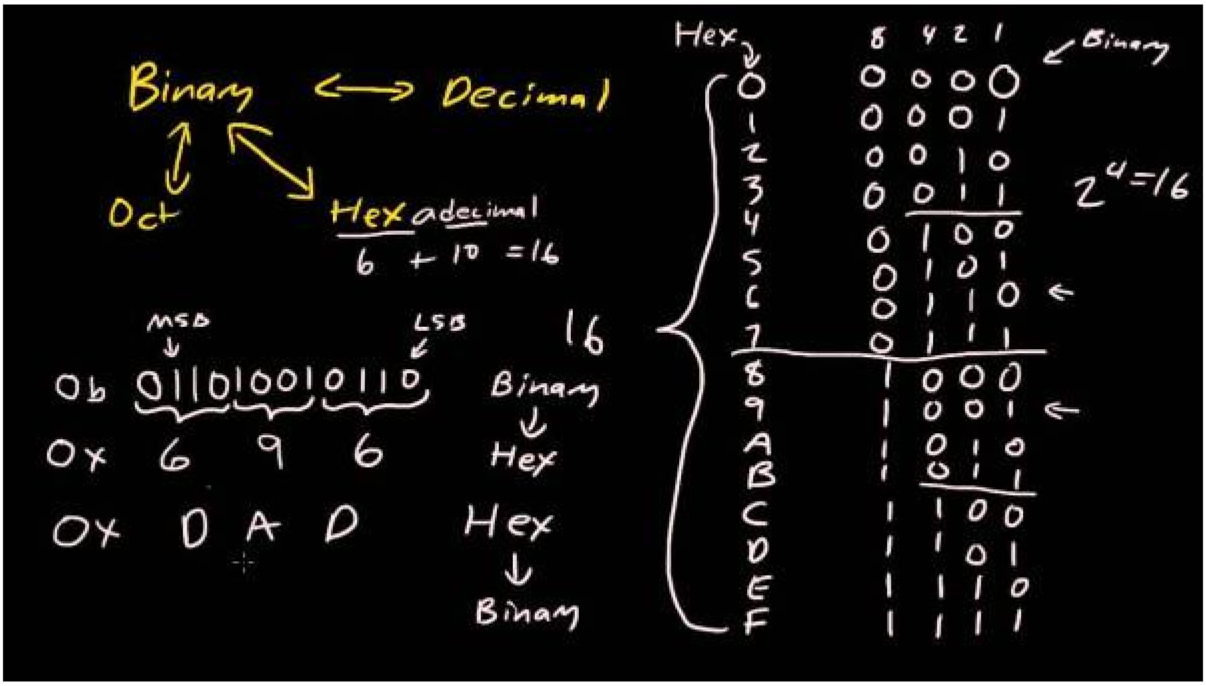
Eliminamos los 0 a la izquierda y queda ##110100101##$_2$
Realizar el pasaje de este numero a decimal es mas sencillo, dado que es mas fácil trabajar en base 2

Otro ejemplo, ahora con decimales



El numero que obtenemos es ##100110101111.000111##$_2$, en este caso eliminamos los 0 a la derecha, después de la coma.

Con números en octal procedemos de forma similar pero representando cada dígito octal en 3 dígitos binarios.



— *Martha*
Volver

(322)