Numbering Systems

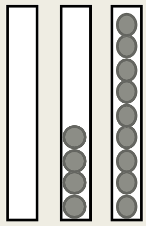
Bring your headphones next Friday

Audio editing

Numbering Systems

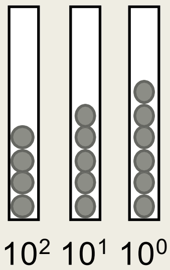
- Decimal (Denary) base 10.
- Other systems
  - *Binary*
  - *Octal*
  - *Hexadecimal*
- Convenient when dealing with computers

Decimal



- Familiar system.
- Count 0 – 9 in any column, gives 10 separate values (a column: a container that holds 9 stones).
- When a count exceeds 9 we start a new column.
- Example:
  - *9 + 1 = 10*
  - *99 + 1 =100*
- Mathematically each column represents a power of ten (Base 10).

Decimal



- 456_{10}= 4 x10^2 + 5 x10^1 + 6 x 10^0
- We call:
  - *10^0 Units*
  - *10^1 Tens*
  - *10^2 Hundreds (x 10)*
  - *………. And so on.*

Binary

2^2 2^1 2^0

- Computers don't have ten fingers and they work in binary bits*.
- A bit can only have 2 values
- 0 and 1
- So when we exceed 1 we must start a new column as we did in decimal.

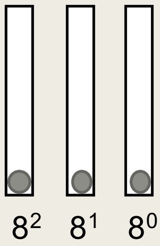
*Quantum computing studies theoretical computation systems (quantum computers) that make direct use of quantum-mechanical phenomena, such as superposition and entanglement, to perform operations on data.[1] Quantum computers are different from digital computers based on transistors. Whereas digital computers require data to be encoded into binary digits (bits), each of which is always in one of two definite states (0 or 1), quantum computation uses quantum bits (qubits), which can be in superpositions of states.
https://en.wikipedia.org/wiki/Quantum_computing

Binary

2^2 2^1 2^0

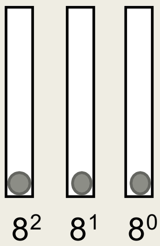
- Examples :
  - $1_2 + 1_2 = 10_2$
  - $0_2 + 1_2 = 1_2$
  - $10111_2 + 11_2 = 11010_2$
- Binary works in powers of 2. (x 2)
  - *Columns are 128 64 32 16 8 4 2 1*
  - $101_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$
    - $2^2 = 4_{10}$
    - $2^1 = 2_{10}$
    - $2^0 = 1_{10}$
- Binary number 101_2 is therefore 5_{10} in decimal.

Octal



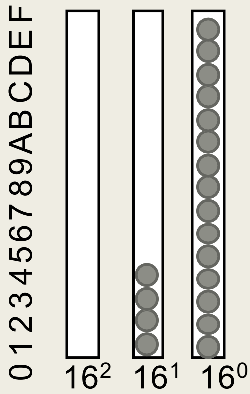
- For large numbers Binary is cumbersome to humans.
- Is there a compromise?
- It groups 3 binary columns together.
- One is octal.
- Octal has 8 values 0 – 7.

Octal



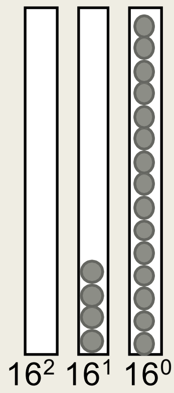
- When we reach 7 we start a new column.
- Examples :
  - *$17_8 + 1_8 = 20_8$*
  - *$777_8 + 1_8 = 1000_8$*
- Octal works in powers of 8. (x 8)
  - *Columns are … 512 64 8 1*
  - *$111_8 = 1 \times 8^2 + 1 \times 8^1 + 1 \times 8^0$*
    - $8^2 = 64$
    - $8^1 = 8$
    - $8^0 = 1$
- Octal number 111_8 is therefore 73_{10} in decimal.

Hexadecimal



- For large numbers Binary is cumbersome to humans.
- Is there a better compromise?
- It groups 4 binary columns together.
- Hexadecimal has 16 values 0 – F. (0 –15 in decimal)
- We need to add 6 more symbols to represent the 2 column numbers 10 –15. We use A – F.
- The prefix "0x" is often used: 0x2AF3

Hexadecimal



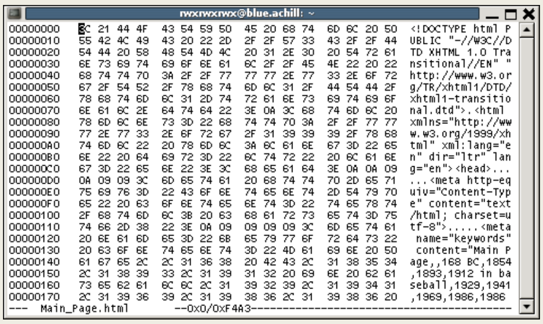
- When we reach F we start a new column.
- Examples :
  – *$1F_{16} + 1_{16} = 20_{16}$*
  – *$FFF_{16} + 1_{16} = 1000_{16}$*
- Hexadecimal works in powers of 16.
  – *Columns are … 4096 256 16 1*
  – *$111 = 1 \times 16^2 + 1 \times 16^1 + 1 \times 16^0$*
- $16^2 = 256$
- $16^1 = 16$
- $16^0 = 1$
- Hexadecimal number 111_{16} is therefore 273_{10} in decimal.

How does Hexadecimal help?

- Consider the 4 bit (column) binary number
  - *$1111_2 = 15_{10} = F_{16}$*
- That is, the largest number that can be represented by 4 bits in binary can be represented by a single value F_{16}.
- All the values in between are then 0 – F.
- Octal does the same for 3 bit binary.

How does Hexadecimal help?

- Example the 16 bit number:
  - *$0010\ 1111\ 0001\ 0100_2$ is $2F14_{16}$.*
- Each hexadecimal digit represents four binary bits as a human-friendly representation.
- Many modern computer systems access data in units of **byte** (8 bits). Byte values can range from 0_{10} to 255_{10}, or two hex digits (range 00_{16} to FF_{16}).



https://en.wikipedia.org/wiki/Hex_editor#/media/File:Hexedit-screenshot.png

Conversions

- Binary, octal, Hex(adecimal) to decimal:
  - *Add up powers as we did earlier.*
- Examples

$111_8 = 1 \times 8^2 + 1 \times 8^1 + 1 \times 8^0 = 64_{10} + 8_{10} + 1_{10} = 73$

$111_{16} = 1 \times 16^2 + 1 \times 16^1 + 1 \times 16^0 = 256_{10} + 16_{10} + 1_{10}$

Conversions

- Decimal to binary.
  - Find the largest power of 2 (column in binary number) that is less than or equal to the number we want to convert.
  - Mark a one in that column.
  - Subtract the power of 2 from the number we want to convert.
  - Repeat the above on the remainder.
  - Repeat until there is no remainder.
  - Fill missing columns with zeros.

Conversions

- Decimal to binary.
  - *Example*
- Convert 67 to binary:

128 **64** 32 16 8 4 2 1

1

64 is the largest column <= 67 leaving remainder 67 – 64 = 3

128 **64** 32 16 8 4 **2** 1

1 1

2 is largest column less than or equal to the remainder 3 . Leaving 1.

1 is largest column less than or equal to remainder 1. Leaving no remainder.

128 **64** 32 16 8 4 **2 1**

1 1 1

Fill with zeros.

So 67_{10} is 0100 0011_2

Conversions

Decimal to hexadecimal.

option 1

- *Find the binary number and separate into groups of four. Express each group of 4 in hexadecimal.*
  - **Example 67_{10} is 0100 0011_2 is 43_{16}.**

option 2

- *Find the largest multiple of each power of 16, less than or equal than your number.*
- *Mark the multiple in that column*
- *Repeat for the remainders as before..*

Conversions

- Decimal to Hexadecimal.
  - *Example*
    - Convert 67 to hexadecimal:

4096 256 16 1

4

4 x 16 = 64 is largest multiple less or equal to 67. Remainder 67 – 64 = 3

4096 256 16 1

4 3

3 x 1 is largest multiple less than or equal to the remainder 3.

Leaving no remainder.

Fill with zeros. (not necessary here)

So 67_{10} is 43_{16}

Conversions

- Hexadecimal to binary.
  - Take each hex symbol in turn.
  - Convert (or look up) to decimal.
  - Convert decimal number to binary.
  - Replace hex symbol with 4 bit binary number.
  - Sounds tedious, but only 6 different from decimal so becomes easy.

Binary	Hex	Decimal
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	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Conversions

- Hexadecimal to binary.
  - *Example:*
  - *Convert 2F to binary.*
    - Take the 2.
    - 2 is 2 in decimal.
    - Convert to binary or look up
    - Gives 0010
    - Take the F
    - F is 15 in decimal
    - Convert to binary or look up
    - Gives 1111
  - *So 2F in binary is 0010 1111*

Binary	Hex	Decimal
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	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Conversion Table

Binary	Hex	Decimal
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	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

MathWorks

Convert Decimal to Binary

dec2bin Convert decimal integer to a binary string.

dec2bin(D) returns the binary representation of D as a string.

D must be a non-negative integer smaller than 2^52.

dec2bin(D,N) produces a binary representation with at least N bits.

Example

dec2bin(23) returns '10111'

MathWorks

Convert Binary to Decimal

bin2dec Convert binary string to decimal integer.

X = bin2dec(B) interprets the binary string B and returns in X the equivalent decimal number.

If B is a character array, or a cell array of strings, each row is interpreted as a binary string.

Embedded, significant spaces are removed. Leading spaces are converted to zeros.

Example

bin2dec('010111') returns 23

bin2dec('010 111') also returns 23

bin2dec(' 010111') also returns 23

Convert decimal to hexadecimal

dec2hex Convert decimal integer to hexadecimal string.

dec2hex(D) returns a 2-D string array where each row is the hexadecimal representation of each decimal integer in D.

D must contain non-negative integers smaller than 2^52.

dec2hex(D,N) produces a 2-D string array where each row contains an N digit hexadecimal number.

Example

dec2hex(2748) returns 'ABC'.

Convert hexadecimal to decimal

hex2dec Convert hexadecimal string to decimal integer.

D = hex2dec(H) interprets the hexadecimal string H and returns in D the equivalent decimal number.

If H is a character array or cell array of strings, each row is interpreted as a hexadecimal string.

EXAMPLES:

hex2dec('12B') and hex2dec('12b') both return 299

MathWorks

Convert Decimal to any base

dec2base Convert decimal integer to base B string.

dec2base(D,B) returns the representation of D as a string in base B. D must be a non-negative integer array smaller than 2^52 and B must be an integer between 2 and 36.

dec2base(D,B,N) produces a representation with at least N digits.

Examples

dec2base(23,3) returns '212' % to base 3

dec2base(23,3,5) returns '00212' % to base 3

dec2base(23,8,4) returns '0027' % to base 8 (Octal)

dec2base(255,16,4) returns '00FF' % to base 16 (Hexa)

MathWorks

Convert any Base to Decimal

base2dec Convert base B string to decimal integer.

base2dec(S,B) converts the string number S of base B into its decimal (base 10) equivalent. B must be an integer between 2 and 36. S must represent a non-negative integer value.

If S is a character array, each row is interpreted as a base B string.

Example

base2dec('212',3) returns 23