PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH ECHAHID HAMMA LAKHDAR
UNIVERSITY OF EL-OUED
FACULTY OF TECHNOLOGY
1ST YEAR LMD SCIENCES AND TECHNIQUES

Chapter II :

Coding of information
ترميز المعلومة في الإعلام الآلي

Dr . BERHOUM Adel
professor at University of El Oued

University year 2023 - 2024

Outline

History

1. Introduction
2. The Basics of numbering.
3. Transcoding (Basic conversion)
4. Binary Arithmetic
5. The ASCII code.

Star

History

This is interesting. Number System played a vital role in Computing. We came across the History of Computing, and we have seen that numbers laid the foundation of computing and helped us to renovate new tools for easier and simpler calculation. These are the lists of several number system that were invented at different ages and cultures in History.

هذا مثير للاهتمام. لعب نظام الأرقام دورًا حيويًا في الحوسبة. لقد عثرنا على تاريخ الحوسبة، ورأينا أن الأرقام أرست أساس الحوسبة وساعدتنا على تجديد أدوات جديدة لإجراء عمليات حسابية أسهل وأبسط. هذه هي قوائم العديد من أنظمة الأعداد التي تم اختراعها في مختلف الأعمار والثقافات في التاريخ.

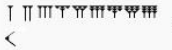
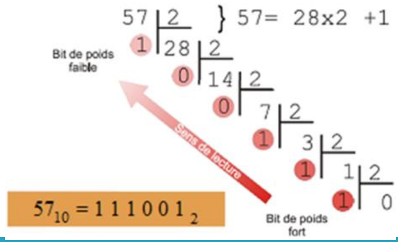
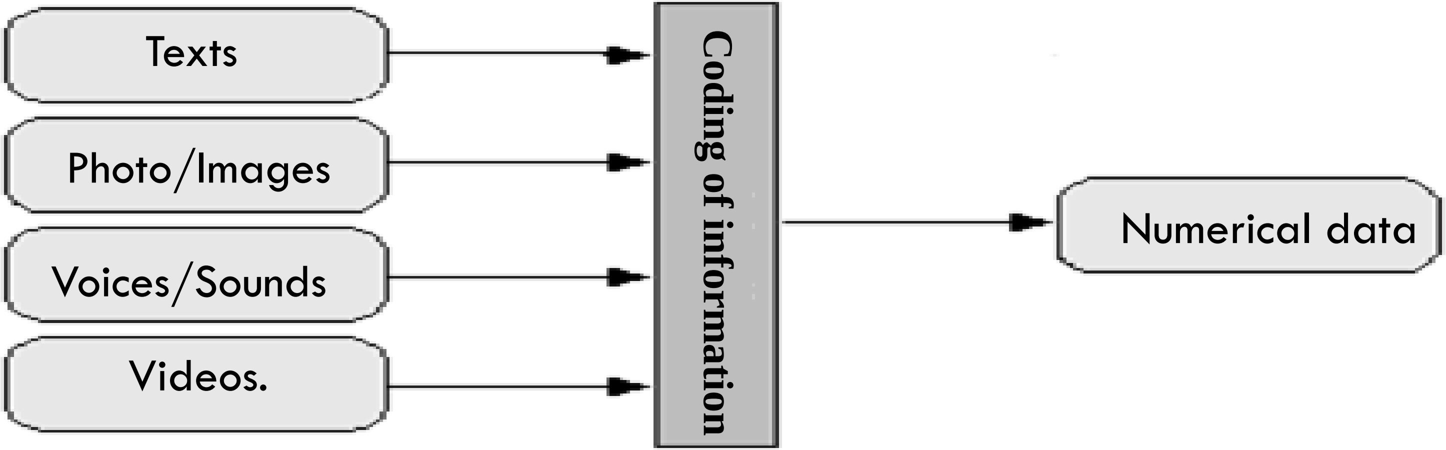
Name	Base	Sample	Approx. first appearance
Babylonian numerals	60		3100 BC
Egyptian numerals	10	 or 	3000 BC
Maya numerals	20		
Oracle bone script	0	0	14th century BC?
Chinese numerals, Japanese numerals, Korean numerals (Sino-Korean)	10	零一二三四五六七八九	
Roman numerals	10	Ⅰ Ⅱ Ⅲ Ⅳ Ⅴ Ⅵ Ⅶ Ⅷ Ⅸ Ⅹ	1000 BC
Greek numerals	10	α β γ δ ε ϝ ζ η θ ι	After 100 BC
Chinese rod numerals	10		1st century
Hindu-Arabic Numerals	10	0 1 2 3 4 5 6 7 8 9	9th century
John Napier's Location arithmetic	2	a b ab c ac bc abc d ad bd	1617 in Rabdology, a non-positional binary system

Diagram of codes (symbols) throughout history

1. Introduction

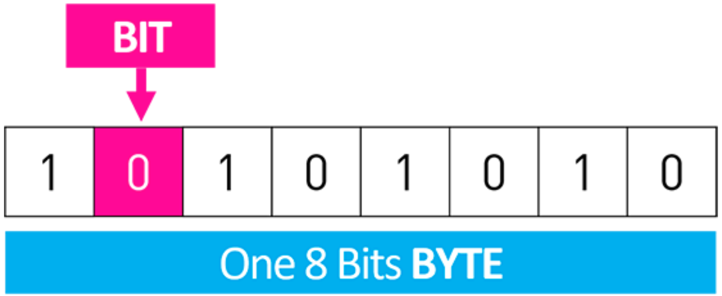


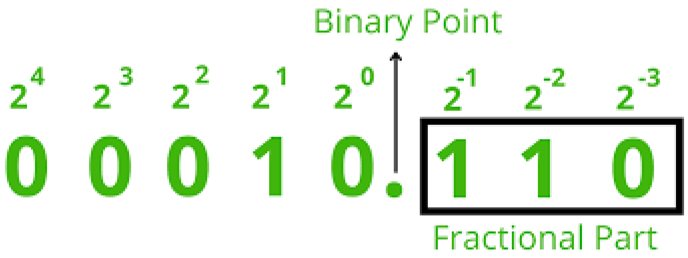
○ Whatever the nature of the information processed by a computer (image, sound, text, video), it is always in the form of a set of numbers written in base 2, for example 01001011. **Coding of information**



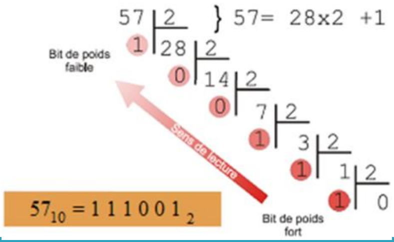
1. Introduction

- The term **bit** (lowercase b in notations) means "**binary digit**", that is to say **0** or **1** in binary numbering.
- It is the smallest unit of information that can be manipulated by a digital machine.

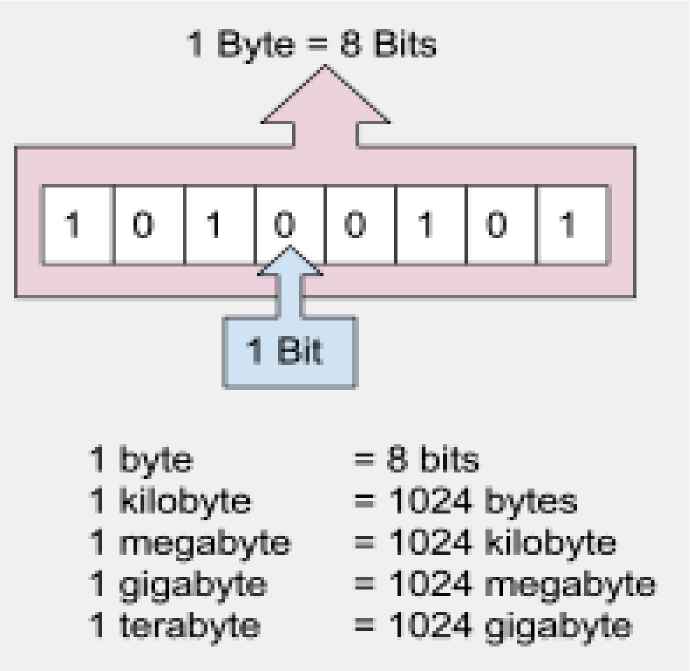




1. Introduction

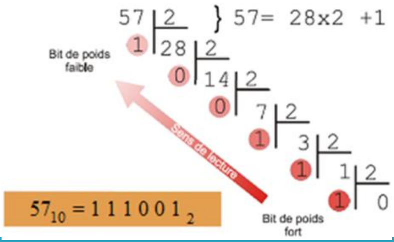


○ The Byte (in English byte or capital B in notations) is a unit of information composed of 8 bits.

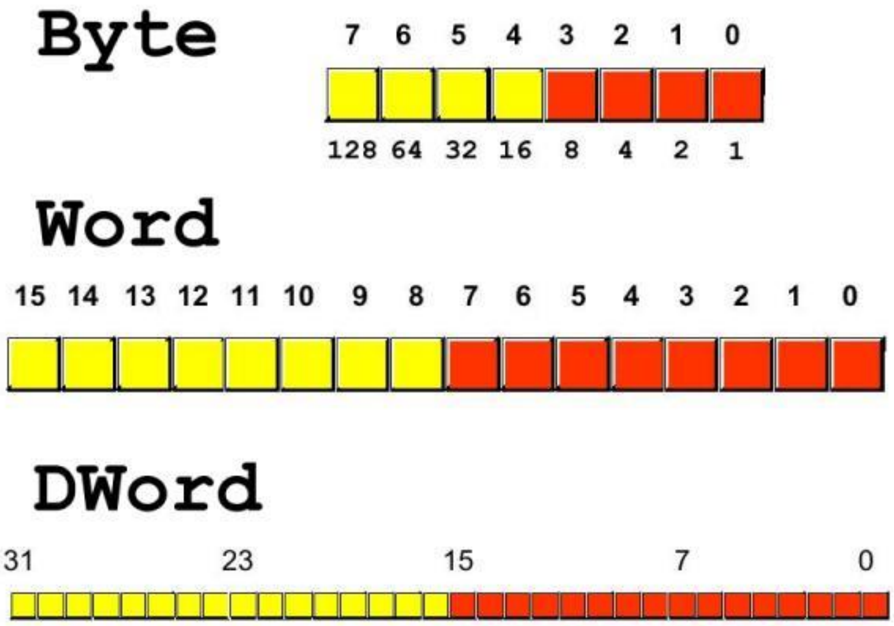


Example, it allows you to store (Save) a character such as a letter or a number.

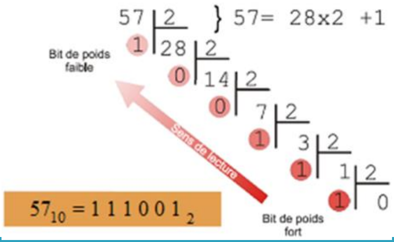
1. Introduction

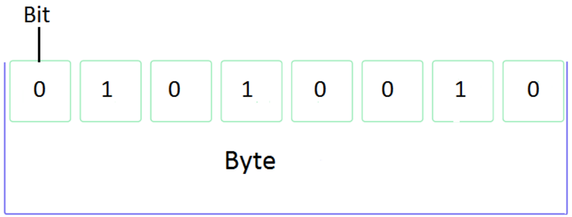


- A unit of information made up of 16 bits is generally called **a word**.
- A unit of information 32 bits long is called **a double word**, hence the name dword.



1. Introduction



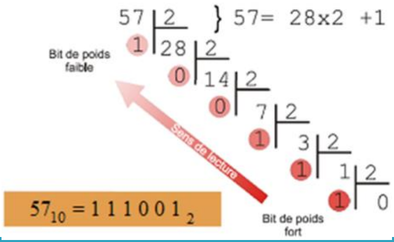


8 bits = 1 Byte

Nom	Symbole	Taille
Kilo-octet	ko	10^3
Méga-octet	Mo	10^6
Giga-octet	Go	10^9
Tera-octet	To	10^{12}
Peta-octet	Po	10^{15}

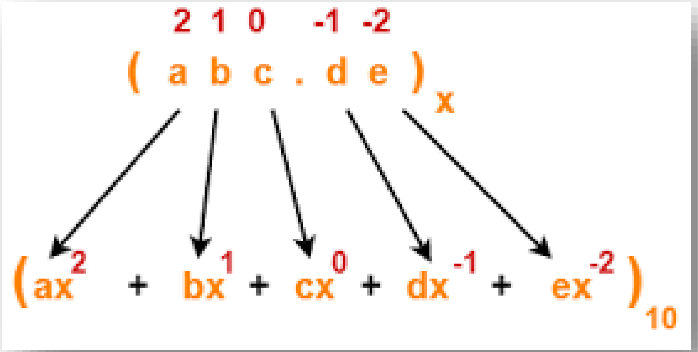
Nom	Symbole	Taille
Kibi-octet	kio	2^{10}
Mébi-octet	Mio	2^{20}
Gibi-octet	Gio	2^{30}
Tebi-octet	Tio	2^{40}
Pebi-octet	Pio	2^{50}

2. The basics of numbering



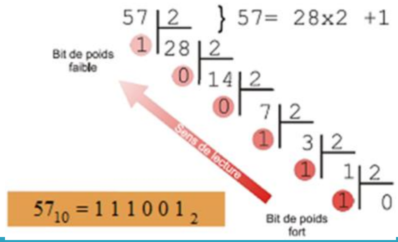
- **Base 10**

General rule



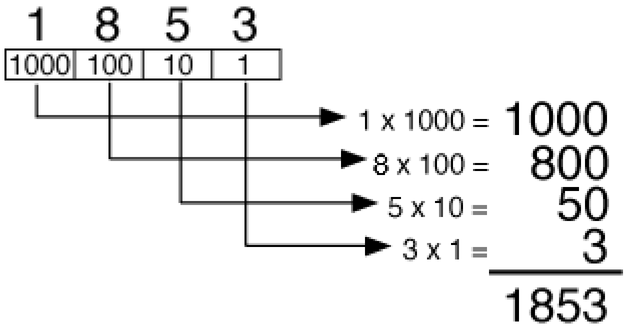
- We use the decimal system (**base 10**) in our **daily activities**. This system is based on ten symbols, from 0 to 9.

2. The basics of numbering

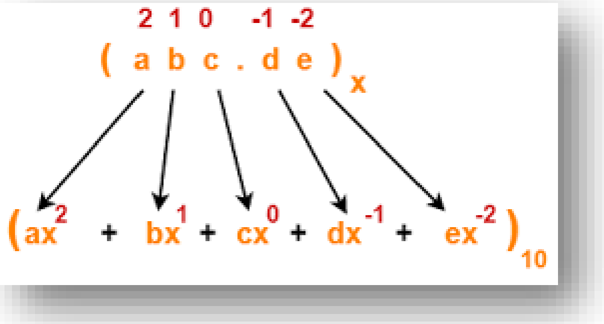


- **Base 10**

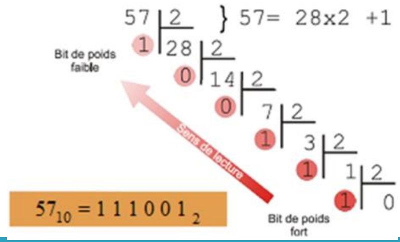
  - Example:



Coding $(2092)_{10}$ = (…………………….)$_{10}$



2. The basics of numbering



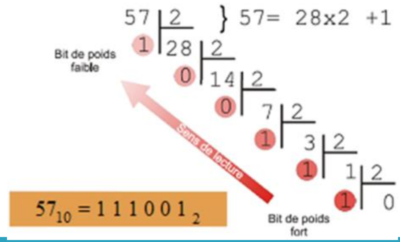
- **Base 2**

○ In computer science, the **binary system** (**base 2**) is very frequently used since **Boolean algebra** is the basis of digital electronics.

○ Two symbols are enough: **0** and **1**.

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
0	0	0	0	0	0	0	0	**0**
128	64	32	16	8	4	2	1	**1**

2. The basics of numbering

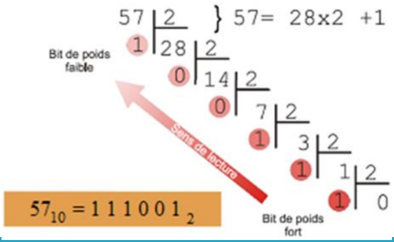


- **Bases (2 – 8 – 16):**

  - Binary system (**b=2**): uses two numbers: {**0,1**} It is with this system that computers work,
  - Octal System (**b=8**): uses eight digits: {**0,1,2,3,4,5,6,7**} Used some time ago in IT. It allows 3 bits to be encoded with a single symbol.
  - Hexadecimal system (**b=16**): uses 16 digits: {**0,1,2,3,4,5,6,7,8,9,A=10, B=11 C=12, D=13,E= 14,F=15**}. This base is widely used in the world of microcomputers. It allows 4 bits to be encoded by a single symbol.

3. Transcoding (Basic conversion)

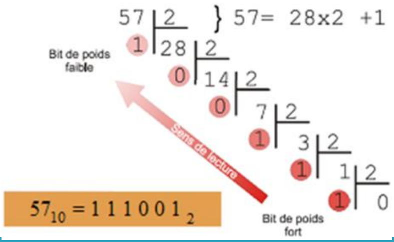
تحويل الترميز



Is the operation which allows you to go from the representation of a number expressed in one base to the representation of the same number but expressed in another base.

هي العملية التي تسمح لك بالانتقال من تمثيل رقم معبر عنه في قاعدة واحدة إلى تمثيل نفس الرقم ولكن معبر عنه في قاعدة أخرى.

3. Transcoding (Basic conversion)



Rule 1 : (From 10 to 2, 8, 16)

Convert from lower order to higher order.

1 .. Mark the number of the number lowest to highest power from 0 to the end of the order (from right to left)

2. Take the lowest order base to the sign power and multiply it by the distinct number of the base.

التحويل من الترتيب الأدنى إلى الترتيب الأعلى.

1. ضع علامة على رقم الرقم الأدنى إلى الأعلى قوة من 0 إلى نهاية الترتيب(من اليمين إلى اليسار)

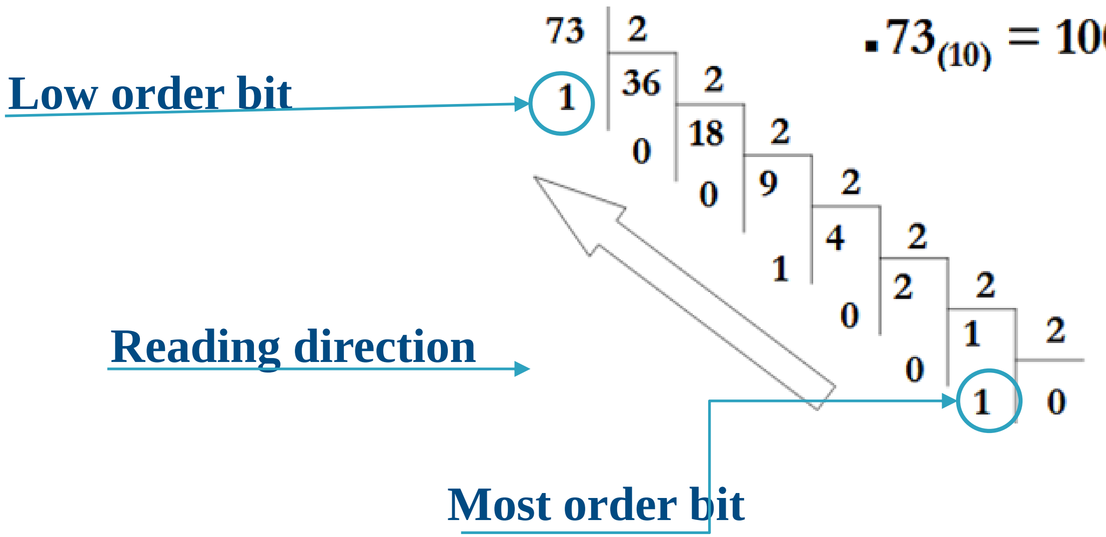
2. خذ قاعدة الترتيب الأدنى إلى قوة العلامة واضربها بالرقم المميز للقاعدة.

BIT#:	7	6	5	4	3	2	1	0
	0	0	1	0	1	1	0	1
VALUE:	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
	128	64	32	16	8	4	2	1

3. Transcoding (Basic conversion)

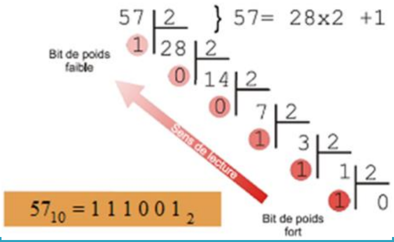
Rule 1: From 10 To 2, 8, 16 :

- Dividing the decimal number by the number b (the target base).
- The condition for stopping the division is to reach the remainder of the quotient 0,
- Arrange the number with the remainder of the division from bottom to top.



$73_{(10)} = 1001001_{(2)}$

3. Transcoding (Basic conversion)

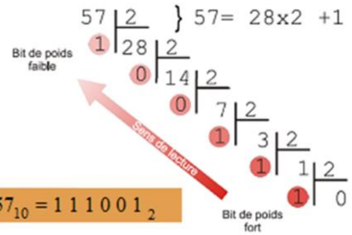


1. Transcoding from base 10 to base b

The rule to follow is successive divisions:

- Then the quotient by the base b
- So on until obtaining a zero quotient
- The sequence of remains corresponds to the symbols of the target base.
- We divide the number by the base b.

3. Transcoding (Basic conversion)

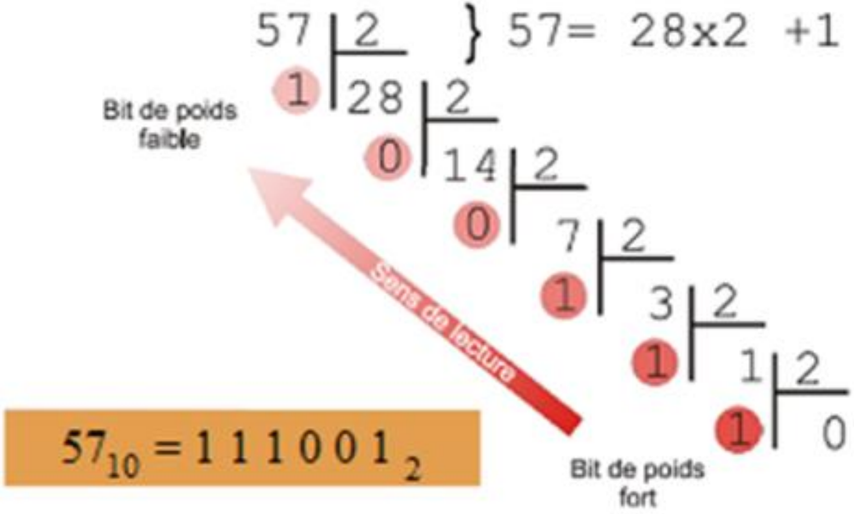


1. Transcoding from base 10 to base b

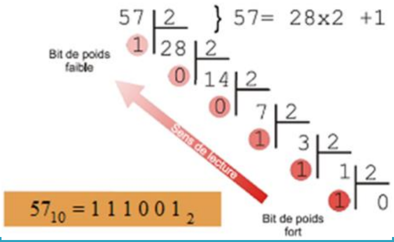
Example1:

Convert decimal number **57** to base 2. (**Decimal to Binary**).

- This involves making a series of Euclidean divisions by 2.
- The result will be the juxtaposition of the remainders.



3. Transcoding (Basic conversion)

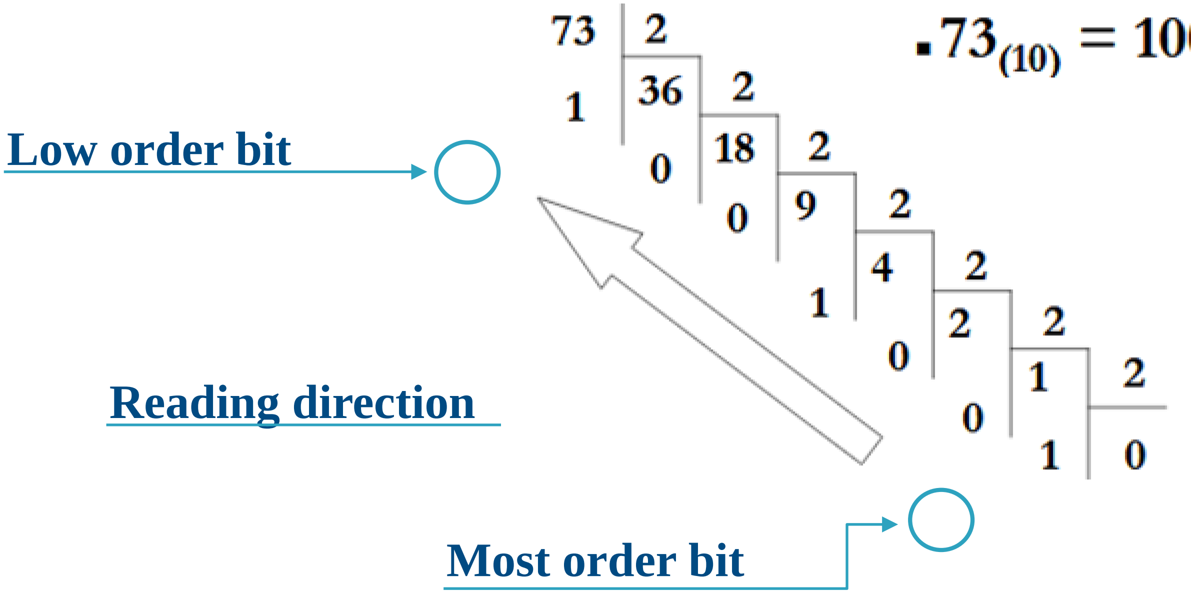


1. Transcoding from base 10 to base b

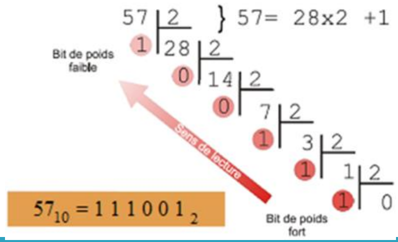
Example 2:

Convert decimal number 73 to base 2. (Decimal to Binary).

- $73_{(10)} = 1001001_{(2)}$



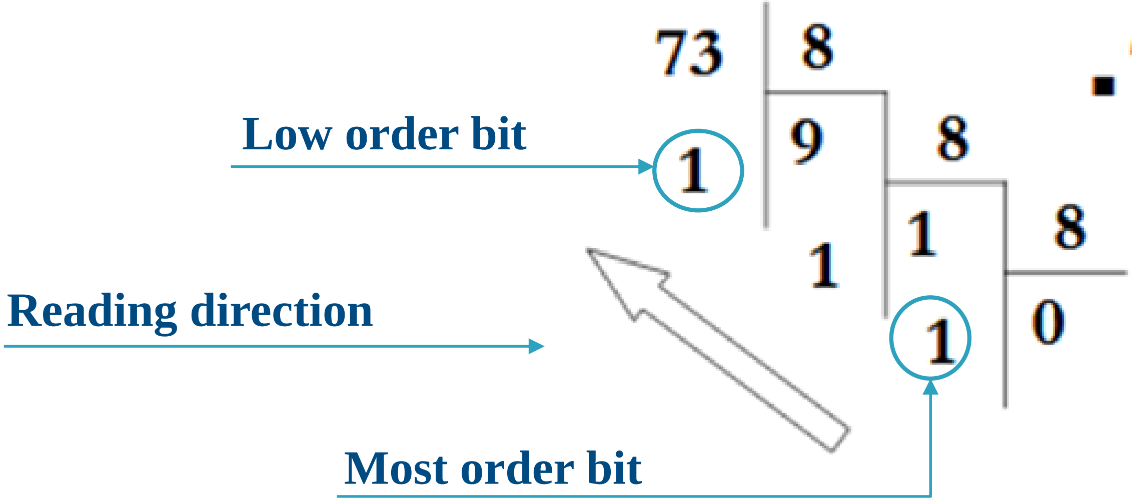
3. Transcoding (Basic conversion)



1. Transcoding from base 10 to base b

Example3:

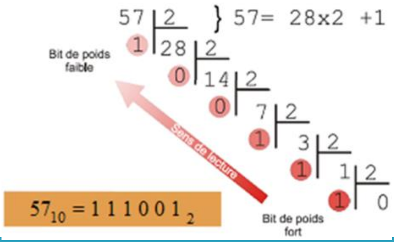
Convert decimal number 73 to base 8. (Decimal to Octel). This involves making a series of Euclidean divisions by 8. The result will be the juxtaposition of the remainders.



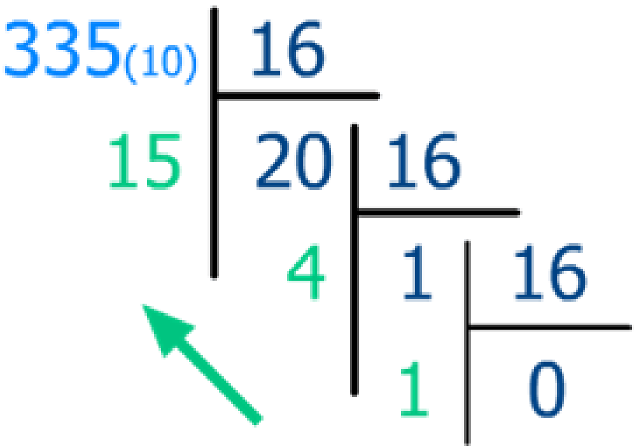
$$73_{(10)} = 111_{(8)}$$

3. Transcoding (Basic conversion)

1. Transcoding from base 10 to base b



Example 4: Convert decimal number 335 to base 16. (Decimal to hexadecimal). This involves making a series of **Euclidean divisions** by 2. The result will be the juxtaposition of the remainders.



$335_{(10)} = 14F_{(16)}$ *(15=F)*

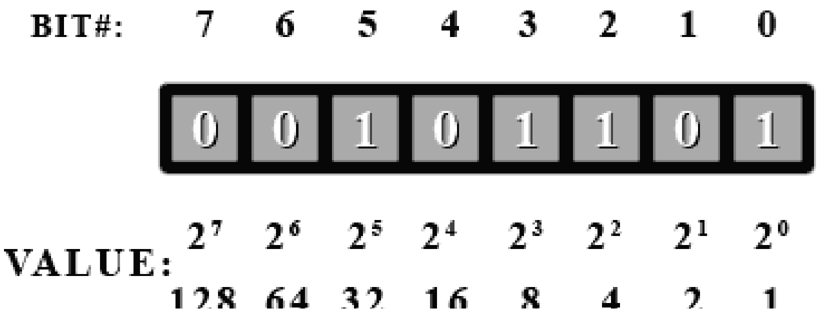
intuitive method
successive subtractions

16^2	*16^1*	*16^0*
256	16	1
1	4	F

3. Transcoding (Basic conversion)

2. From binary base to base b

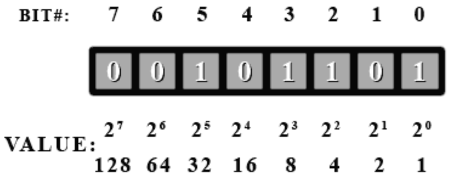
First solution: Convert the number in binary base to decimal base then convert this number in base 10 to base b.



Ex

- $10010_{(2)} = ?_{(8)}$
- $10010_{(2)} = 2^4+2_{(10)}=18_{(10)}=2*8^1+2*8^0{}_{(10)}=22_{(8)}$

3. Transcoding (Basic conversion)



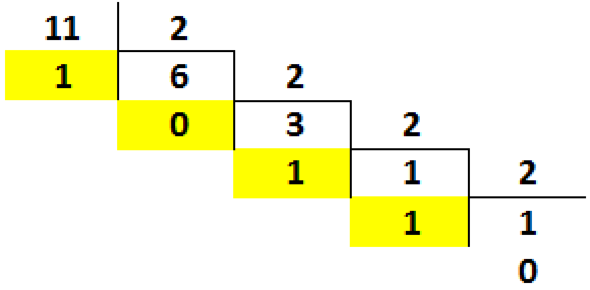
2. From binary base to base b

First solution: Convert the number in binary base to decimal base then convert this number in base 10 to base b.

$$1011_{(2)} = (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0)$$

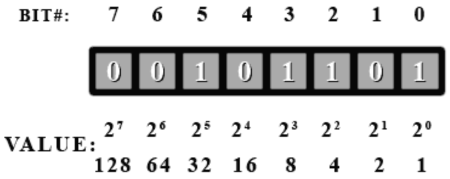
$$1011_{(2)} = (1 \times 8) + (0 \times 4) + (1 \times 2) + (1 \times 1)$$

$$1011_{(2)} = 11_{(10)}$$



$(11)_{10} = (10011)_2$

3. Transcoding (Basic conversion)



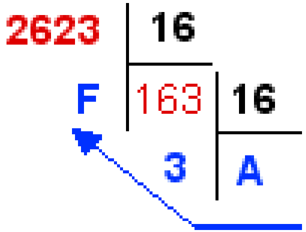
3. From a base to base b

First solution: Convert the number in binary base to decimal base then convert this number in base 10 to base b.

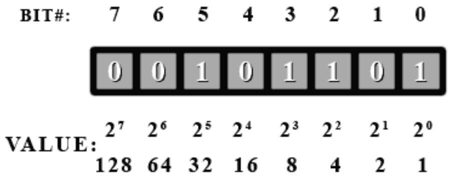
$$A3F_{(16)} = (A \times 16^{2}) + (3 \times 16^{1}) + (F \times 16^{0})$$

$$A3F_{(16)} = (10 \times 256) + (3 \times 16) + (15 \times 1)$$

$$A3F_{(16)} = 2560 + 48 + 15 = 2623_{(10)}$$

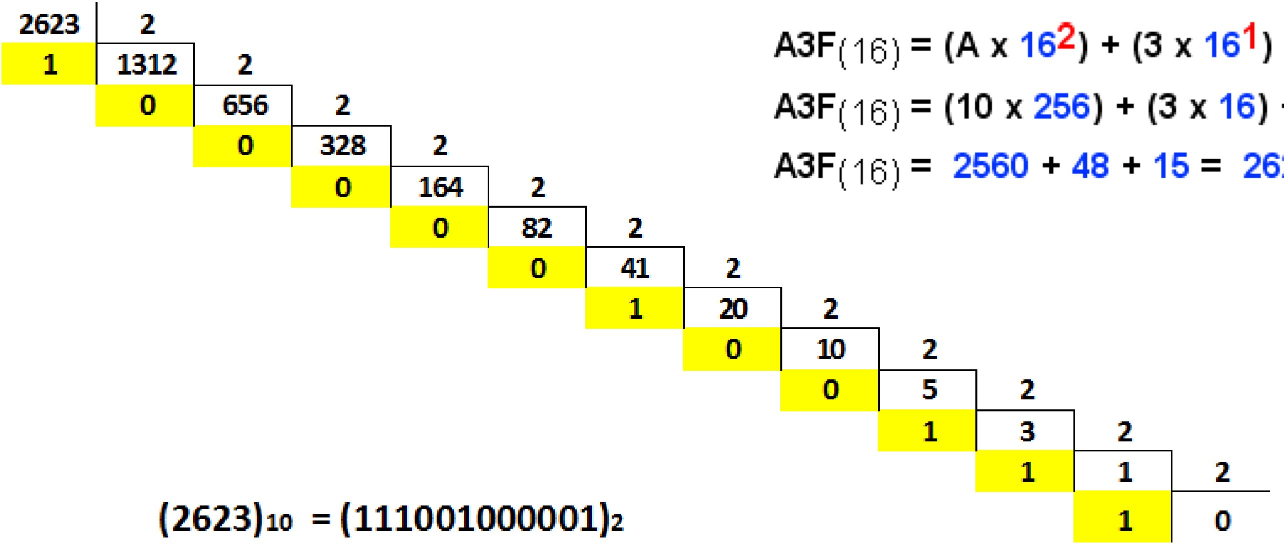


3. Transcoding (Basic conversion)



4. From binary base to base b

First solution: Convert the number in binary base to decimal base then convert this number in base 10 to base b.



$(2623)_{10} = (111001000001)_2$

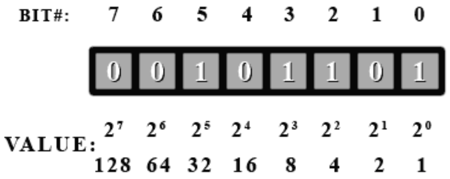
$A3F_{(16)} = (A \times 16^2) + (3 \times 16^1) + (F \times 16^0)$

$A3F_{(16)} = (10 \times 256) + (3 \times 16) + (15 \times 1)$

$A3F_{(16)} = 2560 + 48 + 15 = 2623_{(10)}$

3. Transcoding (Basic conversion)

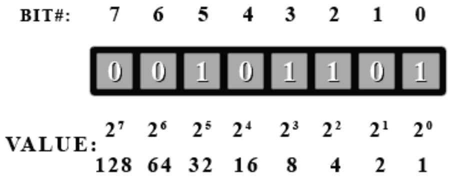
4. From binary base to base b



Second solution:

- Binary to octal: grouping the bits into subsets of three bits then replacing each group with the corresponding symbol in base 8. (Table_1)
- Binary to Hexadecimal: grouping the bits into subsets of four bits then replacing each group with the corresponding symbol in base 16. (Table_2).
- Convert the number in binary base to decimal base then convert this number in base 10 to base b.

3. Transcoding (Basic conversion)



4. From binary base to base b

Second solution:

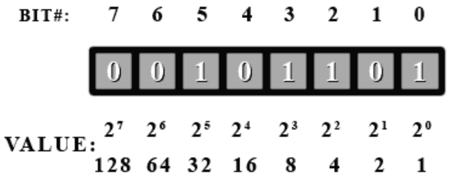
Binary to octal: grouping the bits into subsets of three bits then replacing each group with the corresponding symbol in base 8. (Table_1)

Correspondance Octale \Binaire

Symbole Octale	suite binaire
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

(Table_1)

3. Transcoding (Basic conversion)



4. From binary base to base b

Deuxième solution :

- **Binary to Hexadecimal**:

Grouping the bits into subsets of four bits then replacing each group with the corresponding symbol in base 16. (Table_2). Convert the number in binary base to decimal base then convert this number in base 10 to base b.

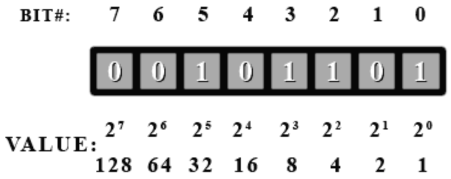
Correspondance Hexadécimale \Binaire

Hexadécimale\Binaire

S. Hexad.	suite binaire	S. Hexad.	suite binaire
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111

(Table_2)

3. Transcoding (Basic conversion)



4. From binary base to base b

Correspondence table

BASE 10	BASE 2 (16)	BASE 16	BASE (8)
0	0000	0	000
1	0001	1	001
2	0010	2	010
3	0011	3	011
4	0100	4	100
5	0101	5	101
6	0110	6	110
7	0111	7	111
8	1000	8	
9	1001	9	
10	1010	A	
11	1011	B	
12	1100	C	
13	1101	D	
14	1110	E	
15	1111	F	

(Table_3)

3. Transcoding (Basic conversion)

BIT#:	7	6	5	4	3	2	1	0
	0	0	1	0	1	1	0	1
VALUE:	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
	128	64	32	16	8	4	2	1

4. From binary base to base b

Examples:

4			2			1		
1	0	0	0	1	0	0	0	1
8	7	6	5	4	3	2	1	0
2^8				2^4				2^0
256				16				1
273								

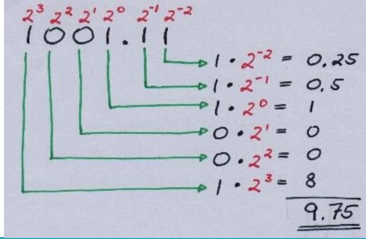
A62 = 5142

A				6				2			
1	0	1	0	0	1	1	0	0	0	1	0
5			1			4			2		

7015 = E0D

7			0			1			5		
1	1	1	0	0	0	0	0	1	1	0	1
E				0				D			

4. Coding of real numbers and signed numbers:



1. Real numbers

- A real number is represented in decimal by: $d_m d_{m-1} \ldots d_1 d_0 . d_{-1} d_{-2} \ldots d_{-n}$

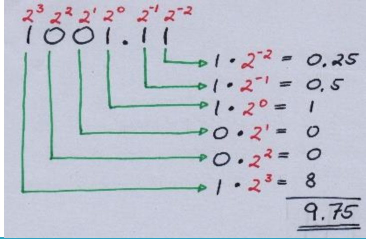
To go from a decimal real number to binary we use the following method:

1-Division the entire part by 2.

2-Multiply the fractional part by 2.

- In binary, a real number is represented by the following form: $b_m b_{m-1} \ldots b_1 b_0 . b_{-1} b_{-2} \ldots b_{-n}$

4. Coding of real numbers and signed numbers:

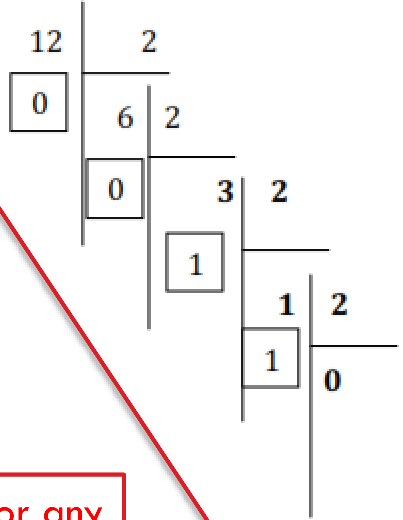


1. Real numbers

Examples 01:

$(12.55)_2 = (1100.1000110...)_2$

La partie entière $(12)_{10} = (1100)_2$



The fractional part $(0.50)_{10} = (1000110...)_2$

0.55×2= 1.10

0.10×2= 0.2

0.2×2= 0.4

0.4×2= 0.8

0.8 ×2= 1.6

0.6×2= 1.2

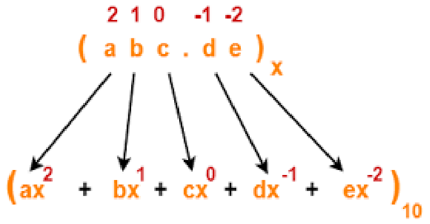
0.2×2= 0.4

.

.

We define a similar notation for any binary number representing a real

4. Coding of real numbers and signed numbers:



1. Real numbers

Examples 02: the number **(1010.101)** represents the following sum**:**

$$1010{,}101 = 8 + 0 + 2 + 0 + \frac{1}{2} + \frac{0}{4} + \frac{1}{8}$$

Here we divide by powers of 2 instead of powers of 10 in decimal. Or, in decimal writing:

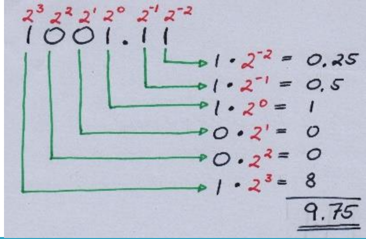
8 + 2+ 0.5 + 0.125 = 10.625

In another form:

$$1 \times 2^{3} + 0 \times 2^{2} + 1 \times 2^{1} + 0 \times 2^{0} + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$$

4. Coding of real numbers and signed numbers:

2. Signed numbers



Coding signed numbers

To go from a negative number in **decimal to binary**, simply calculate:

The 1's complement and the 2's complement.

$(17)_{10}$ = $(0\ 00\ 10001)_2$

1's complement of 1

11101110

2's complement of 2

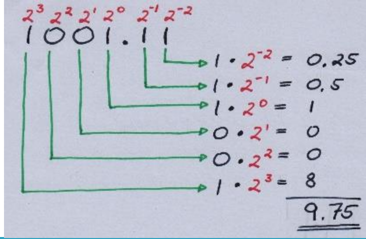
11101110+
1

11101111 = $(-17)_{10}$

Bit 1 indicates that the sign is negative

5. Binary Arithmetic

Binary arithmetic is essential part of all the digital computers and many other digital system.



Binary Addition

It is a key for binary subtraction, multiplication, division. There are four rules of binary addition.

Case	A + B	Sum	Carry
1	0 + 0	0	0
2	0 + 1	1	0
3	1 + 0	1	0
4	1 + 1	0	1

In fourth case, a binary addition is creating a sum of (0 .e.i (10 =1 +1 is written in the given column and a carry of 1 over to the next column.

Example – Addition

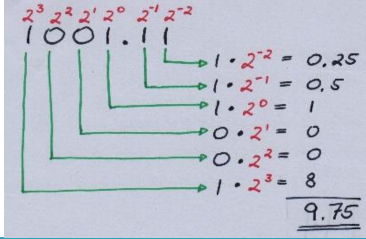
0011010 + 001100 = 00100110

```
  1 1              carry
  0 0 1 1 0 1 0   = 26₁₀
+ 0 0 0 1 1 0 0   = 12₁₀
  -------------
  0 1 0 0 1 1 0   = 38₁₀
```

https://www.tutorialspoint.com/computer_logical_organization/binary_arithmetic.htm

5. Binary Arithmetic

Binary arithmetic is essential part of all the digital computers and many other digital system.



Binary Subtraction

Subtraction and Borrow, these two words will be used very frequently for the binary subtraction. There are four rules of binary subtraction.

Case	A - B	Subtract	Borrow
1	0 - 0	0	0
2	1 - 0	1	0
3	1 - 1	0	0
4	0 - 1	0	1

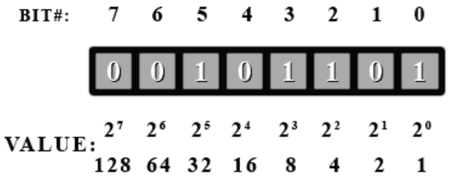
Binary Subtraction

0011010 - 001100 = 00001110

1 1	borrow
0 0 ~~1~~ ~~1~~ 0 1 0	$= 26_{10}$
- 0 0 0 1 1 0 0	$= 12_{10}$
0 0 0 1 1 1 0	$= 14_{10}$

https://www.tutorialspoint.com/computer_logical_organization/binary_arithmetic.htm

6. ASCII Code



The ASCII standard (usually pronounced “aski”) allows all kinds of machines to store, analyze and communicate textual information. In particular, almost all personal computers and workstations use ASCII encoding.

Codes 0 to 31 are not characters. They are called control characters because they allow actions such as: carriage return (13).

Codes 32 to 64 represent symbols such as: the symbol + is associated with (43).

Codes 65 to 90 represent capital letters such as: the character A is associated with (65).

Codes 97 to 122 represent lowercase letters such as: the character a is associated with (97)

6. ASCII Code

The following table contains all the printable characters of the basic ASCII standard :

Character	Decimal Number	Binary Number	Character	Decimal Number	Binary Number
blank space	32	0010 0000	^	94	0101 1110
!	33	0010 0001	-	95	0101 1111
"	34	0010 0010	`	96	0110 0000
#	35	0010 0011	a	97	0110 0001
$	36	0010 0100	b	98	0110 0010
A	65	0100 0001	c	99	0110 0011
B	66	0100 0010	d	100	0110 0100
C	67	0100 0011	e	101	0110 0101
D	68	0100 0100	f	102	0110 0110
E	69	0100 0101	g	103	0110 0111
F	70	0100 0110	h	104	0110 1000
G	71	0100 0111	i	105	0110 1001
H	72	0100 1000	j	106	0110 1010
I	73	0100 1001	k	107	0110 1011
J	74	0100 1010	l	108	0110 1100
K	75	0100 1011	m	109	0110 1101
L	76	0100 1100	n	110	0110 1110
M	77	0100 1101	o	111	0110 1111
N	78	0100 1110	p	112	0111 0000
O	79	0100 1111	q	113	0111 0001
P	80	0101 0000	r	114	0111 0010
Q	81	0101 0001	s	115	0111 0011
R	82	0101 0010	t	116	0111 0100
S	83	0101 0011	u	117	0111 0101
T	84	0101 0100	v	118	0111 0110
U	85	0101 0101	w	119	0111 0111
V	86	0101 0110	x	120	0111 1000
W	87	0101 0111	y	121	0111 1001
X	88	0101 1000	z	122	0111 1010
Y	89	0101 1001	{	123	0111 1011
Z	90	0101 1010	|	124	0111 1100
[	91	0101 1011	}	125	0111 1101
/	92	0101 1100	~	126	0111 1110
]	93	0101 1101			

6. ASCII Code

Dec	Chr	Dec	Chr	Dec	Chr	Dec	Chr	Dec	Chr	Dec	Chr	Dec	Chr
32		46	.	60	<	74	J	88	X	102	f	115	t
33	!	47	/	61	=	75	K	89	Y	103	g	116	u
34	"	48	0	62	>	76	L	90	Z	104	h	117	v
35	#	49	1	63	?	77	M	91	[	105	i	118	w
36	$	50	2	64	@	78	N	92	\	106	j	119	x
37	%	51	3	65	A	79	O	93	]	107	k	120	y
38	&	52	4	66	B	80	P	94	^	108	l	121	z
39	'	53	5	67	C	81	Q	95	_	109	m		
40	(	54	6	68	D	82	R	96	`	111	n		
41	)	55	7	69	E	83	S	97	a	110	o		
42	*	56	8	70	F	84	T	98	b	111	p		
43	+	57	9	71	G	85	U	99	c	112	q		
44	,	58	:	72	H	86	V	100	d	113	r		
45	-	59	;	73	I	87	W	101	e	114	s		

The following table contains all the printable characters of the basic ASCII standard :

Dec	Hx	Oct	Char	
0	0	000	NUL	(null)
1	1	001	SOH	(start of heading)
2	2	002	STX	(start of text)
3	3	003	ETX	(end of text)
4	4	004	EOT	(end of transmission)
5	5	005	ENQ	(enquiry)
6	6	006	ACK	(acknowledge)
7	7	007	BEL	(bell)
8	8	010	BS	(backspace)
9	9	011	TAB	(horizontal tab)
10	A	012	LF	(NL line feed, new line)
11	B	013	VT	(vertical tab)
12	C	014	FF	(NP form feed, new page)
13	D	015	CR	(carriage return)
14	E	016	SO	(shift out)
15	F	017	SI	(shift in)
16	10	020	DLE	(data link escape)
17	11	021	DC1	(device control 1)
18	12	022	DC2	(device control 2)
19	13	023	DC3	(device control 3)
20	14	024	DC4	(device control 4)
21	15	025	NAK	(negative acknowledge)
22	16	026	SYN	(synchronous idle)
23	17	027	ETB	(end of trans. block)
24	18	030	CAN	(cancel)
25	19	031	EM	(end of medium)
26	1A	032	SUB	(substitute)
27	1B	033	ESC	(escape)
28	1C	034	FS	(file separator)
29	1D	035	GS	(group separator)
30	1E	036	RS	(record separator)
31	1F	037	US	(unit separator)

Dec	Hx	Oct	Html	Chr
32	20	040		Space
33	21	041	!	!
34	22	042	"	"
35	23	043	#	#
36	24	044	$	$
37	25	045	%	%
38	26	046	&	&
39	27	047	'	'
40	28	050	(	(
41	29	051	)	)
42	2A	052	*	*
43	2B	053	+	+
44	2C	054	,	,
45	2D	055	-	-
46	2E	056	.	.
47	2F	057	/	/
48	30	060	0	0
49	31	061	1	1
50	32	062	2	2
51	33	063	3	3
52	34	064	4	4
53	35	065	5	5
54	36	066	6	6
55	37	067	7	7
56	38	070	8	8
57	39	071	9	9
58	3A	072	:	:
59	3B	073	;	;
60	3C	074	<	<
61	3D	075	=	=
62	3E	076	>	>
63	3F	077	?	?

Dec	Hx	Oct	Html	Chr
64	40	100	@	@
65	41	101	A	A
66	42	102	B	B
67	43	103	C	C
68	44	104	D	D
69	45	105	E	E
70	46	106	F	F
71	47	107	G	G
72	48	110	H	H
73	49	111	I	I
74	4A	112	J	J
75	4B	113	K	K
76	4C	114	L	L
77	4D	115	M	M
78	4E	116	N	N
79	4F	117	O	O
80	50	120	P	P
81	51	121	Q	Q
82	52	122	R	R
83	53	123	S	S
84	54	124	T	T
85	55	125	U	U
86	56	126	V	V
87	57	127	W	W
88	58	130	X	X
89	59	131	Y	Y
90	5A	132	Z	Z
91	5B	133	[	[
92	5C	134	\	\
93	5D	135	]	]
94	5E	136	^	^
95	5F	137	_	_

Dec	Hx	Oct	Html	Chr
96	60	140	`	`
97	61	141	a	a
98	62	142	b	b
99	63	143	c	c
100	64	144	d	d
101	65	145	e	e
102	66	146	f	f
103	67	147	g	g
104	68	150	h	h
105	69	151	i	i
106	6A	152	j	j
107	6B	153	k	k
108	6C	154	l	l
109	6D	155	m	m
110	6E	156	n	n
111	6F	157	o	o
112	70	160	p	p
113	71	161	q	q
114	72	162	r	r
115	73	163	s	s
116	74	164	t	t
117	75	165	u	u
118	76	166	v	v
119	77	167	w	w
120	78	170	x	x
121	79	171	y	y
122	7A	172	z	z
123	7B	173	{	{
124	7C	174			|
125	7D	175	}	}
126	7E	176	~	~
127	7F	177	&#127;	DEL

Source : www.LookupTables.com

Extended ASCII Codes

128	Ç	144	É	160	á	176	░	192	└	208	╨	224	α	240	≡
129	ü	145	æ	161	í	177	▒	193	┴	209	╤	225	ß	241	±
130	é	146	Æ	162	ó	178	▓	194	┬	210	╥	226	Γ	242	≥
131	â	147	ô	163	ú	179	│	195	├	211	╙	227	π	243	≤
132	ä	148	ö	164	ñ	180	┤	196	─	212	╘	228	Σ	244	⌠
133	à	149	ò	165	Ñ	181	╡	197	┼	213	╒	229	σ	245	⌡
134	å	150	û	166	ª	182	╢	198	╞	214	╓	230	µ	246	÷
135	ç	151	ù	167	º	183	╖	199	╟	215	╫	231	τ	247	≈
136	ê	152	ÿ	168	¿	184	╕	200	╚	216	╪	232	Φ	248	°
137	ë	153	Ö	169	⌐	185	╣	201	╔	217	┘	233	Θ	249	·
138	è	154	Ü	170	¬	186	║	202	╩	218	┌	234	Ω	250	·
139	ï	155	¢	171	½	187	╗	203	╦	219	█	235	δ	251	√
140	î	156	£	172	¼	188	╝	204	╠	220	▄	236	∞	252	ⁿ
141	ì	157	¥	173	¡	189	╜	205	═	221	▌	237	φ	253	²
142	Ä	158	₧	174	«	190	╛	206	╬	222	▐	238	ε	254	■
143	Å	159	ƒ	175	»	191	┐	207	╧	223	▀	239	∩	255	

Source : www.LookupTables.com

32		46	.	60	<	74	J	88	X	102	f	115	t
33	!	47	/	61	=	75	K	89	Y	103	g	116	u
34	"	48	0	62	>	76	L	90	Z	104	h	117	v
35	#	49	1	63	?	77	M	91	[	105	i	118	w
36	$	50	2	64	@	78	N	92	\	106	j	119	x
37	%	51	3	65	A	79	O	93	]	107	k	120	y
38	&	52	4	66	B	80	P	94	^	108	l	121	z
39	'	53	5	67	C	81	Q	95	_	109	m		
40	(	54	6	68	D	82	R	96	`	111	n		
41	)	55	7	69	E	83	S	97	a	110	o		
42	*	56	8	70	F	84	T	98	b	111	p		
43	+	57	9	71	G	85	U	99	c	112	q		
44	,	58	:	72	H	86	V	100	d	113	r		
45	-	59	;	73	I	87	W	101	e	114	s		

6. ASCII Code

ASCII to Binary, Part I

Using the ASCII conversion chart (see handout) and the Binary/Hexadecimal table above, answer the following questions.

1. What is the hexadecimal representation for the string ""?
2. What is the binary representation for the string "Hello"?

1. Hello = H e l l o = (72 101 108 108 110)$_{10}$ = (48 65 6C 6C 6E)
2. Hello = H e l l o = (72 101 108 108 110)$_{10}$ = (01001000 01100101 01101100 01101100 01101110)

https://www.rapidtables.com/convert/number/decimal-to-binary.html

6. ASCII Code

ASCII to Binary, Part I

32		46	.	60	<	74	J	88	X	102	f	115	t
33	!	47	/	61	=	75	K	89	Y	103	g	116	u
34	"	48	0	62	>	76	L	90	Z	104	h	117	v
35	#	49	1	63	?	77	M	91	[	105	i	118	w
36	$	50	2	64	@	78	N	92	\	106	j	119	x
37	%	51	3	65	A	79	O	93	]	107	k	120	y
38	&	52	4	66	B	80	P	94	^	108	l	121	z
39	'	53	5	67	C	81	Q	95	_	109	m		
40	(	54	6	68	D	82	R	96	`	111	n		
41	)	55	7	69	E	83	S	97	a	110	o		
42	*	56	8	70	F	84	T	98	b	111	p		
43	+	57	9	71	G	85	U	99	c	112	q		
44	,	58	:	72	H	86	V	100	d	113	r		
45	-	59	;	73	I	87	W	101	e	114	s		

Consider the following binary code :

1. Express the binary code above in hexadecimal (all on one line) .
2. If these numbers represent an ASCII string, what string do they represent ?

		d					
		100					
01000111	01101111	01100100	00100000	01110000	01110010	01101111	01110100

				G				
				71				
01100101	01100011	01110100	00100000	01000111	01100001	01111001	01100001	00100001

6. ASCII Code

ASCII to Binary, Part I

32		46	.	60	<	74	J	88	X	102	f	115	t
33	!	47	/	61	=	75	K	89	Y	103	g	116	u
34	"	48	0	62	>	76	L	90	Z	104	h	117	v
35	#	49	1	63	?	77	M	91	[	105	i	118	w
36	$	50	2	64	@	78	N	92	\	106	j	119	x
37	%	51	3	65	A	79	O	93	]	107	k	120	y
38	&	52	4	66	B	80	P	94	^	108	l	121	z
39	'	53	5	67	C	81	Q	95	_	109	m		
40	(	54	6	68	D	82	R	96	`	111	n		
41	)	55	7	69	E	83	S	97	a	110	o		
42	*	56	8	70	F	84	T	98	b	111	p		
43	+	57	9	71	G	85	U	99	c	112	q		
44	,	58	:	72	H	86	V	100	d	113	r		
45	-	59	;	73	I	87	W	101	e	114	s		

Solution :

God protect Gaza!

G	o	d		p	r	o	t
71	110	100	32	111	113	110	**115**
01000111	01101111	01100100	00100000	01110000	01110010	01101111	**01110100**

e	c	t		G	a	z	a	!
101	99	115	32	71	97	121	71	**33**
01100101	01100011	01110100	00100000	01000111	01100001	01111001	01100001	**00100001**

6. ASCII Code

ASCII to Binary, Part II

32		46	.	60	<	74	J	88	X	102	f	115	t
33	!	47	/	61	=	75	K	89	Y	103	g	116	u
34	"	48	0	62	>	76	L	90	Z	104	h	117	v
35	#	49	1	63	?	77	M	91	[	105	i	118	w
36	$	50	2	64	@	78	N	92	\	106	j	119	x
37	%	51	3	65	A	79	O	93	]	107	k	120	y
38	&	52	4	66	B	80	P	94	^	108	l	121	z
39	'	53	5	67	C	81	Q	95	_	109	m		
40	(	54	6	68	D	82	R	96	`	111	n		
41	)	55	7	69	E	83	S	97	a	110	o		
42	*	56	8	70	F	84	T	98	b	111	p		
43	+	57	9	71	G	85	U	99	c	112	q		
44	,	58	:	72	H	86	V	100	d	113	r		
45	-	59	;	73	I	87	W	101	e	114	s		

Think of a word or short phrase in English, encode it in binary, and then send it to someone else to decode. Your word or phrase must be at least six letters long; Don't make it too long, either for your sake or for the person who will be decoding it. Make sure it is appropriate to give it to anyone else in the class to decode.

فكر في كلمة أو عبارة قصيرة باللغة الإنجليزية، وقوم بتشفيرها بالنظام الثنائي، ثم أرسلها لشخص آخر لفك تشفيرها. يجب أن تتكون كلمتك أو عبارتك من ستة أحرف على الأقل؛ لا تجعلها طويلة، سواء من أجلك أو من أجل الشخص الذي سيقوم بفك شفرتها. تأكد من أنه من المناسب إعطاؤها لأي شخص آخر في الفصل لفك التشفير.

6. ASCII Code

ASCII to Binary, Part II

32		46	.	60	<	74	J	88	X	102	f	115	t
33	!	47	/	61	=	75	K	89	Y	103	g	116	u
34	"	48	0	62	>	76	L	90	Z	104	h	117	v
35	#	49	1	63	?	77	M	91	[	105	i	118	w
36	$	50	2	64	@	78	N	92	\	106	j	119	x
37	%	51	3	65	A	79	O	93	]	107	k	120	y
38	&	52	4	66	B	80	P	94	^	108	l	121	z
39	'	53	5	67	C	81	Q	95	_	109	m		
40	(	54	6	68	D	82	R	96	`	111	n		
41	)	55	7	69	E	83	S	97	a	110	o		
42	*	56	8	70	F	84	T	98	b	111	p		
43	+	57	9	71	G	85	U	99	c	112	q		
44	,	58	:	72	H	86	V	100	d	113	r		
45	-	59	;	73	I	87	W	101	e	114	s		

1. Write down your word or phrase in English (on this page).
2. Express your word or phrase in hexadecimal. (Question: if you chose to represent a phrase, how do you represent the spaces between words?)
3. Express your word or phrase in binary.
4. Copy the binary representation of your word or phrase to a new word doc. Put your name somewhere on the word doc as well.
5. Give the word doc (via teams) containing the binary representation of your word or phrase to another student in the class, following the instructor's instructions. (Don't forget to make sure your name is on the word doc.)

1. اكتب كلمتك أو عبارتك باللغة الإنجليزية (في ورقة).
2. عبر عن كلمتك أو عبارة بالنظام السادس عشري. (سؤال: إذا اخترت تمثيل عبارة، فكيف تمثل المسافات بين الكلمات؟)
3. عبر عن كلمتك أو عبارة بالنظام الثنائي.
4. انسخ التمثيل الثنائي لكلمتك أو عبارتك (بالنظام الثنائي) إلى مستند جديد.
5. قم بتسليم الورقة الجديد إلى طالب آخر في الفصل. (لا تنس التأكد من كتابة اسمك على الورقة).

6. ASCII Code

Binary to ASCII, Part II

32		46	.	60	<	74	J	88	X	102	f	115	t
33	!	47	/	61	=	75	K	89	Y	103	g	116	u
34	"	48	0	62	>	76	L	90	Z	104	h	117	v
35	#	49	1	63	?	77	M	91	[	105	i	118	w
36	$	50	2	64	@	78	N	92	\	106	j	119	x
37	%	51	3	65	A	79	O	93	]	107	k	120	y
38	&	52	4	66	B	80	P	94	^	108	l	121	z
39	'	53	5	67	C	81	Q	95	_	109	m		
40	(	54	6	68	D	82	R	96	`	111	n		
41	)	55	7	69	E	83	S	97	a	110	o		
42	*	56	8	70	F	84	T	98	b	111	p		
43	+	57	9	71	G	85	U	99	c	112	q		
44	,	58	:	72	H	86	V	100	d	113	r		
45	-	59	;	73	I	87	W	101	e	114	s		

1. What is the name of the person whose binary representation you are about to decode?
2. Write down the binary representation you received from that person.
3. Express that word or phrase in hexadecimal.
4. Write out the word or phrase in English.

1. ما هو اسم الشخص الذي أنت على وشك فك تشفير تمثيله الثنائي؟
2. اكتب التمثيل الثنائي الذي تلقيته من هذا الشخص.
3. قم بالتعبير عن هذه الكلمة أو العبارة بالنظام الست عشري.
4. اكتب الكلمة أو العبارة باللغة الإنجليزية.

For listening and following the course

Dr. BERHOUM Adel