

$$f = \frac{1}{T}$$

Hz

Le code SIRCS/CNTRL-S

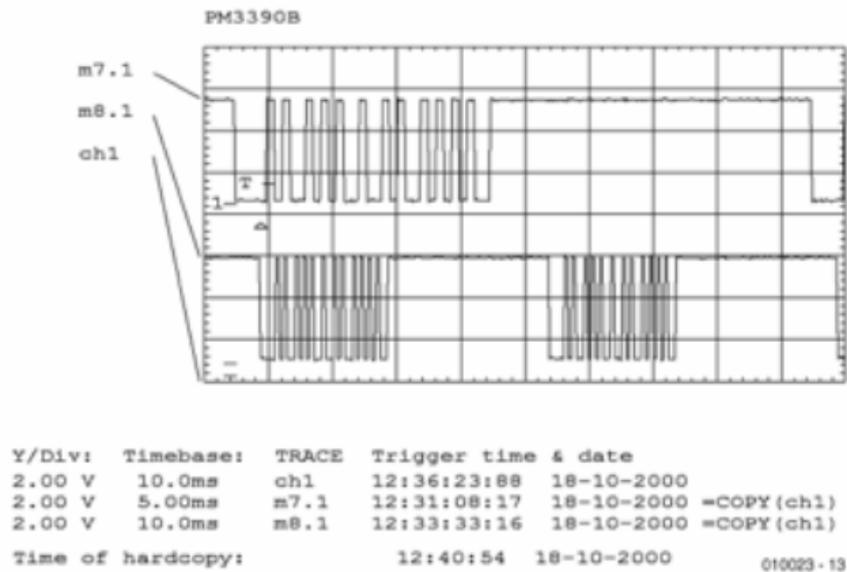


Figure 3. Code SIRCS à la sortie du circuit intégré de réception TFMS5360.

Le protocole SIRCS ou encore CNTRL-S de Sony se compose d'un bit de départ et de 12 à 20 bits d'instruction. La commande est divisée en un code d'adressage de 5 à 13 bits et d'un code de touche de 7 bits. Le déroulement temporel de ces impulsions est indiqué dans la figure 4.

Un long bit de départ (2,4 ms) est tout d'abord émis, suivi d'une pause de 0,6 ms. Les données proprement dites suivent ; 1 est représenté par 1,2 ms actif, 0,6 ms inactif et 0 par 0,6 ms actif et 0,6 ms inactif. La transmission doit être répétée au moins 2 fois (5 fois avec un caméscope) après une pause, on

considère sinon qu'il s'est produit une erreur de transmission. La séquence temporelle de SIRCS correspond à celle de CTRL-S, mais SIRCS est cadencé à 40 kHz. CTRL-S est utilisé pour la commande par fil d'appareils, comme par exemple le couplage de magnétoscopes. L'analyse d'une télécommande de Sony révèle le circuit intégré suivant :

Émetteur : KIE RA275 S42

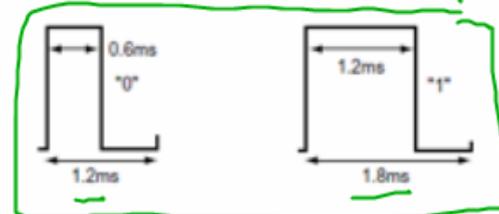
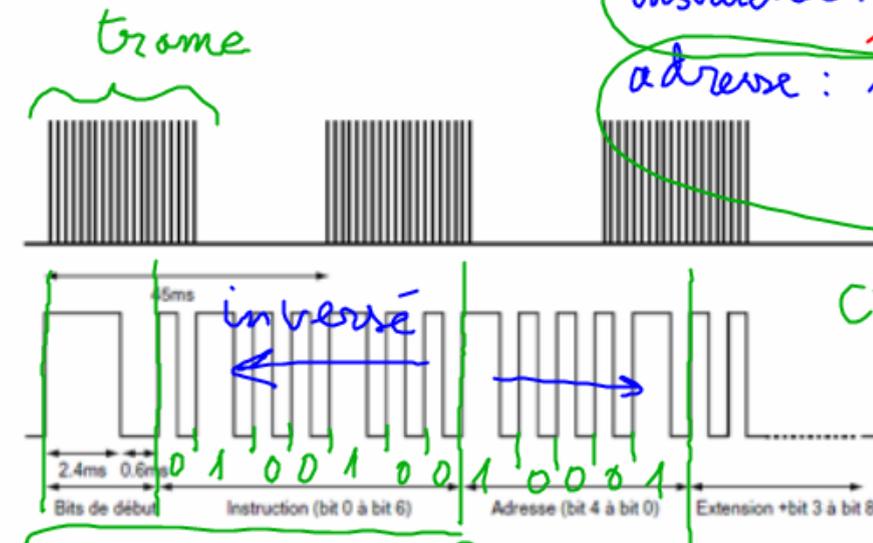
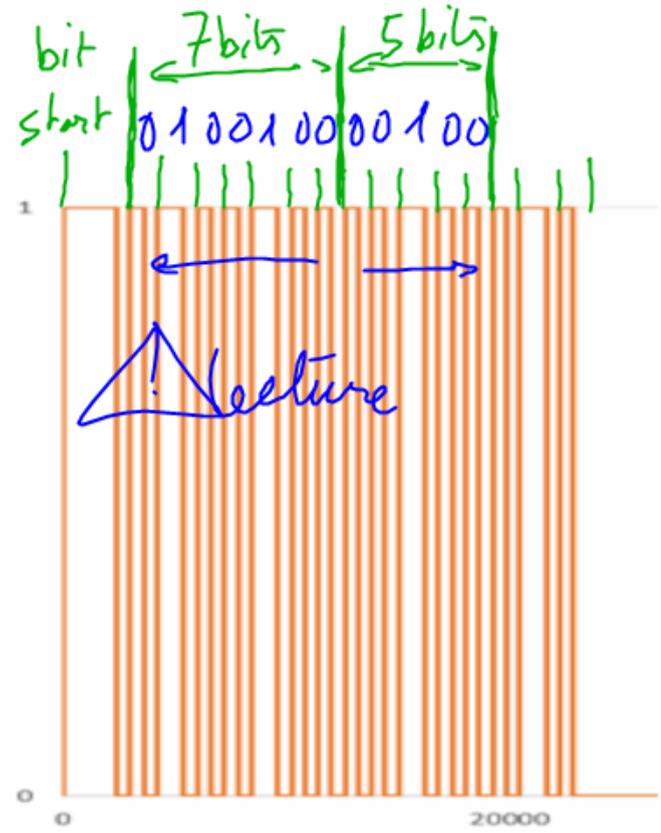
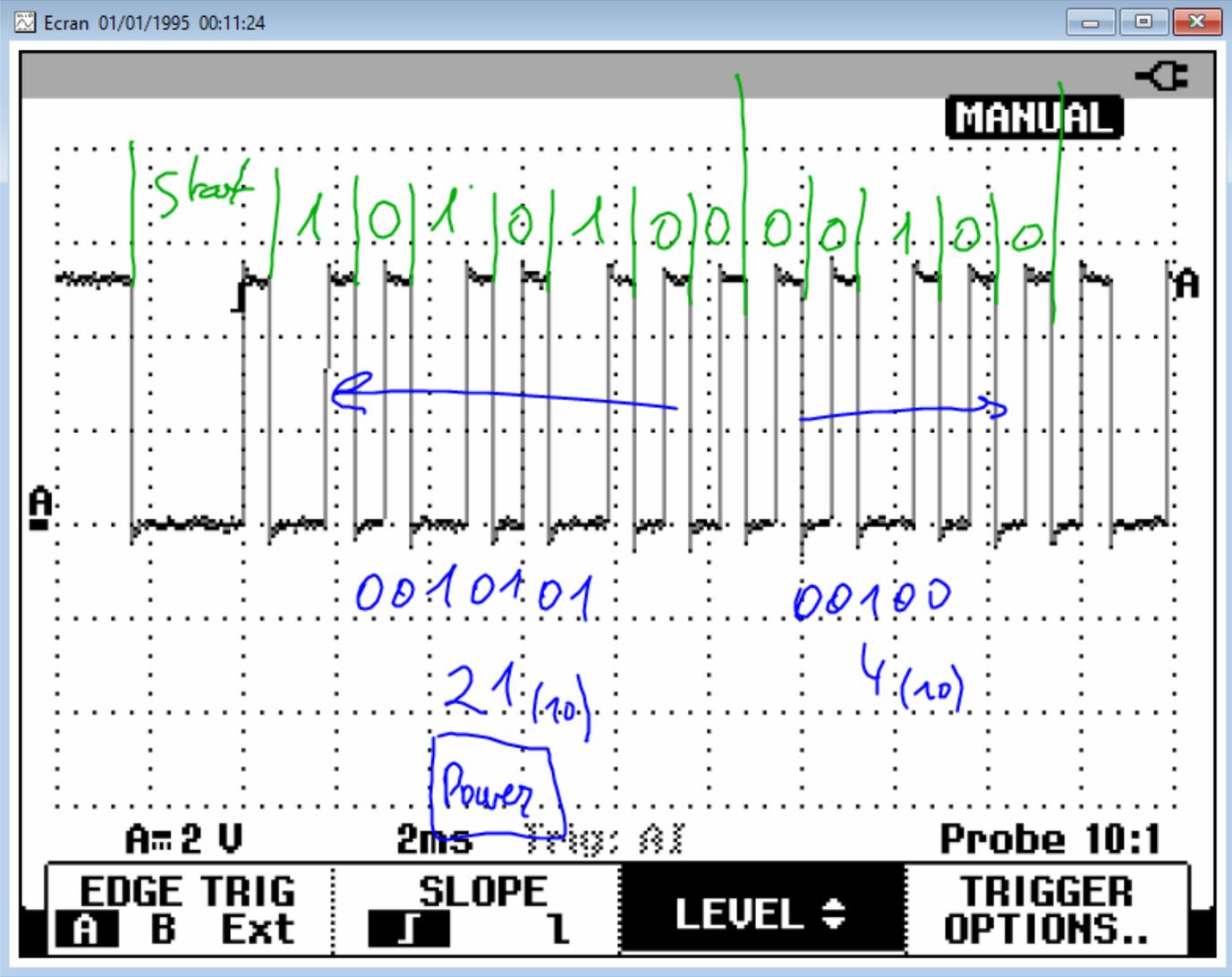
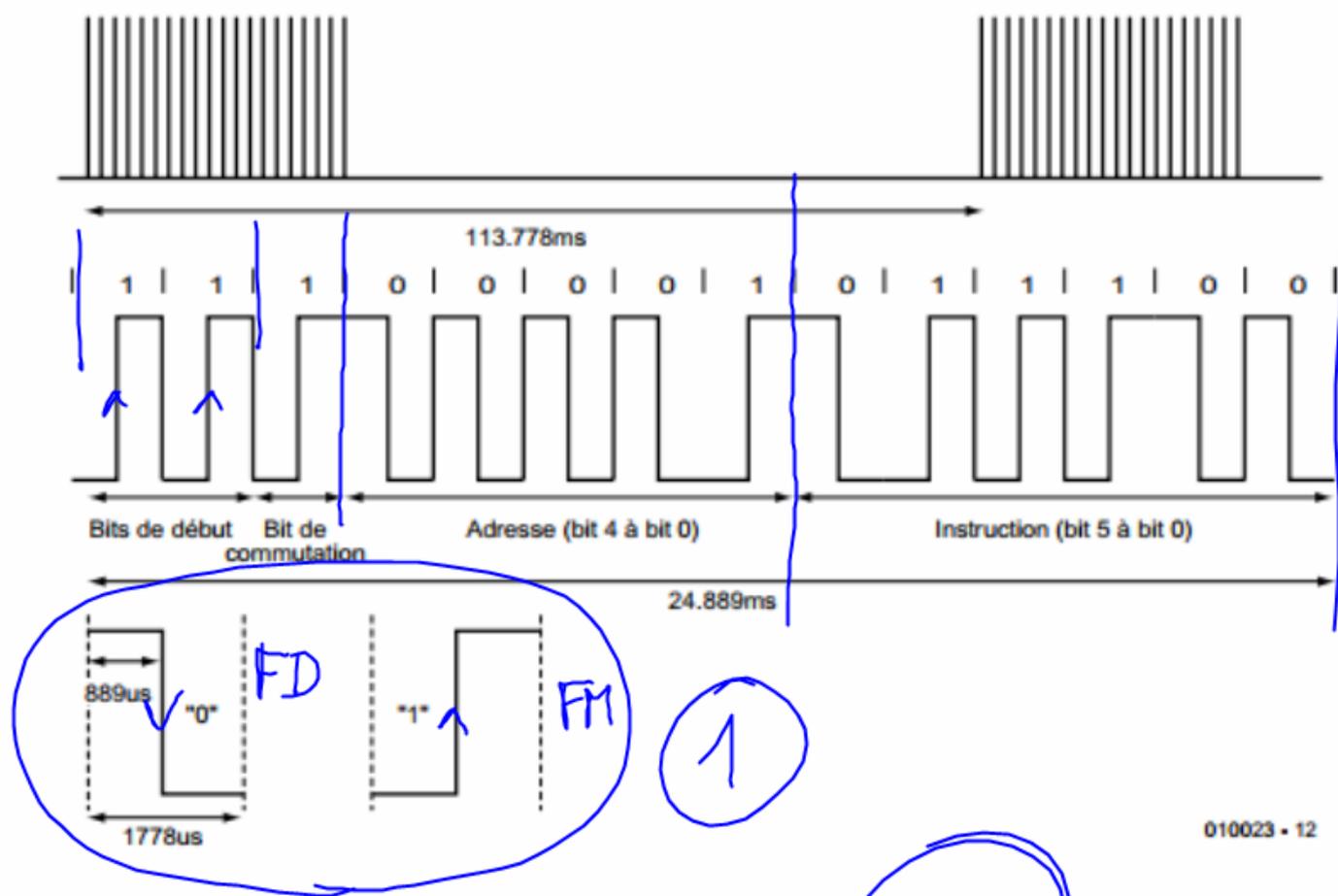


Figure 4. Structure des protocoles CNTRL-S et SIRCS.



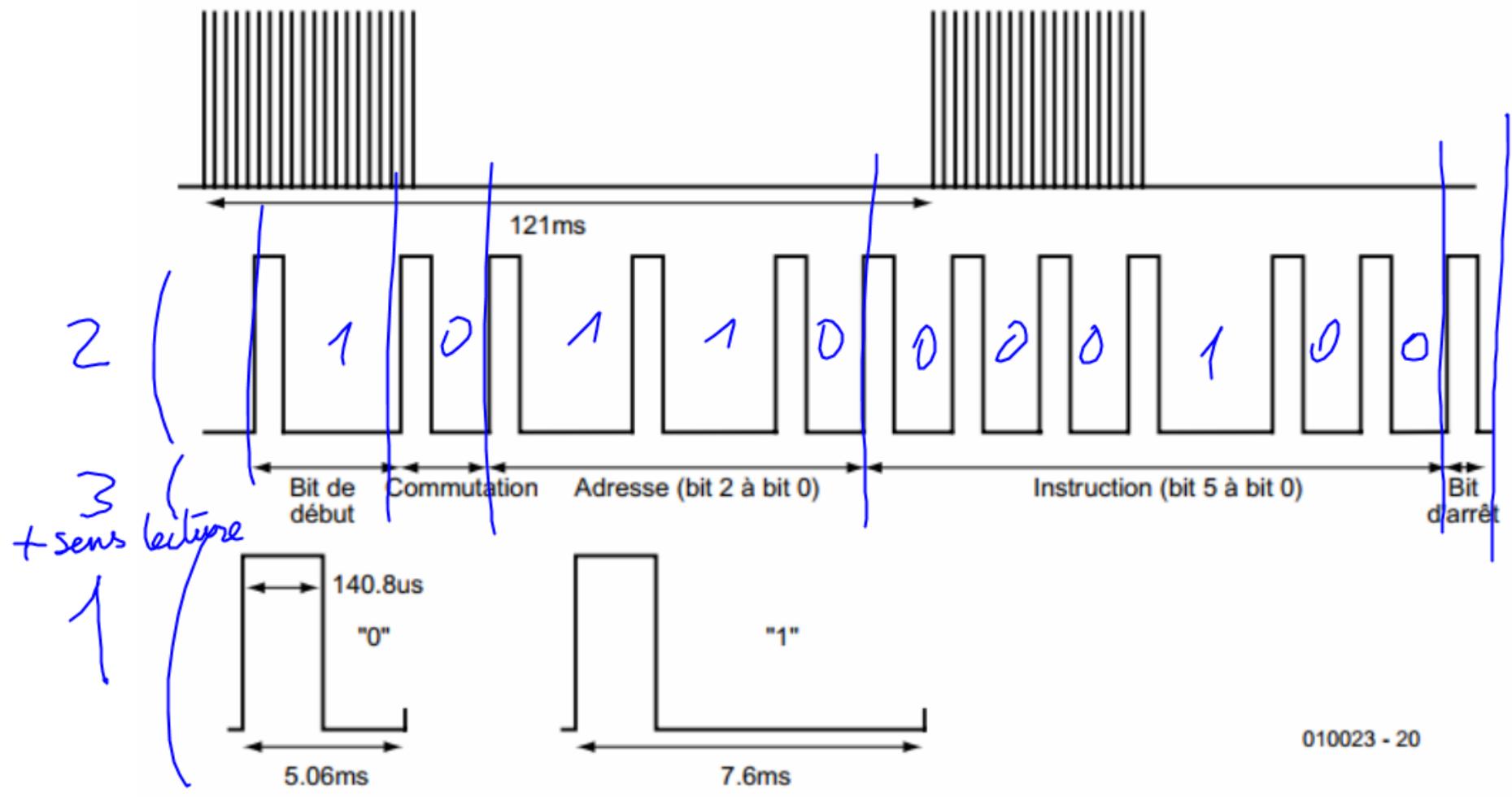
instruct: val +
 $0010010_{(2)} = 18_{(10)}$
 adresse:
 $00100_{(2)} = 4_{(10)}$
 VTR





010023 - 12

Figure 2. Structure du protocole du code RC5 (adresse 1, commande 28 dans cet exemple).



Base 2

0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0

Base 10

Row 2 1 0

0	0
0	0
...	...
1	0
1	1
...	...
1	0
0	1
0	1
0	0

523

2. Représentation d'un nombre

Notation générale

$$\underline{abcd}_n = a \cdot n^3 + b \cdot n^2 + c \cdot n^1 + d \cdot n^0$$

avec n : nombre de chiffres de la base

rang du chiffre

poils du chiffre

8/9

2.1. Notation décimale : 0...9

Exemple avec le nombre $N_{10} = 2923$

N_{10}	2	9	2	3
Rang du chiffre	3	2	1	0
Poids du chiffre	10^3	10^2	10^1	10^0
	1000	100	10	1

On peut généraliser les décompositions de la base 10 par : $abcd_{10} = a \times 10^3 + b \times 10^2 + c \times 10^1 + d \times 10^0$

2.2. Notation binaire : 0-1

Exemple avec le nombre $N_2 = 11010010$

N_2	1	1	0	1	0	0	1	0
Rang du chiffre	7	6	5	4	3	2	1	0
Poids du chiffre	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
	128	64	32	16	8	4	2	1

$$128 + 64 + 16 + 2 = 210_{(10)}$$

2.3. Notation hexadécimale : 0...9-A...F

Exemple avec le nombre $N_{16} = F302$

N_{16}	F	3	0	2
Rang du chiffre	3	2	1	0
Poids du chiffre	16^3	16^2	16^1	16^0

$$15 \cdot 16^3 + 3 \cdot 16^2 + 2 =$$

$$62210_{(10)}$$

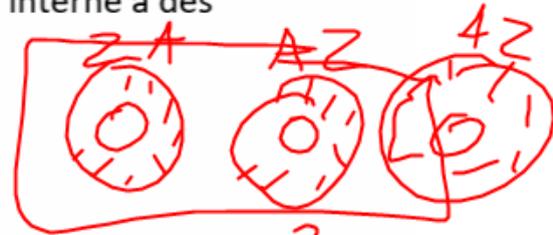
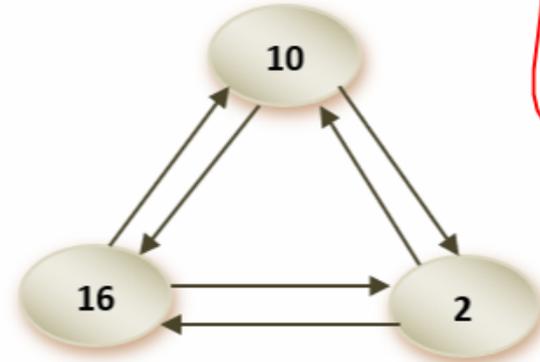
HEX	F302
DEC	62 210
OCT	171 402
BIN	1111 0011 0000 0010

3. Transcodage

Unité de codage : Les composants constituant un système informatique réagissent de manière interne à des signaux « tout ou rien ». Il y a 2 états stables : « 0 » = « L » (low) et « 1 » = « H » (high)
La base 2 est la mieux adaptée, on parle de codage binaire.

Décimal	Binaire naturel	Hexadécimal
0	0000 0000	0
1	0000 0001	1
2	0000 0010	2
3	0000 0011	3
4	0000 0100	4
5	0000 0101	5
6	0000 0110	6
7	0000 0111	7
8	0000 1000	8
9	0000 1001	9
10	0000 1010	A
11	0000 1011	B
12	0000 1100	C
13	0000 1101	D
14	0000 1110	E
15	0000 1111	F
16	0001 0000	10

4 bits $\rightarrow 2^4 = 16$



$$26^3 = 17576$$

- 8 bits = octet (byte)
- 16 bits = octets (word)
- 1 Kio = octets = 1024 octets = ko
- 1 Mio = octets = 1024 Kio = Mo
- 1 Gio = octets = 1024 Mio = Go
- 1 Tio = octets = 1024 Gio = To

Pour plus d'info : <http://fr.wikipedia.org/wiki/Octet>

255 | MMMM FF