THEOPHILE LEGRAND
Chanoine fondateur de l'école en 1937

Lycée Secondaire et Technologique Privé
16, rue Bertrand —BP 32119 LOUVROIL
59606 MAUBEUGE CEDEX
Tél : 03 27 65 52 60 Fax : 03 27 62 14 69

BTS
CRSA

CONCEPTION ET REALISATION
DE
SYSTEMES AUTOMATISES
(CRSA)

Les systèmes de numération et codes

Lycée privé Théophile Legrand	BTS CRSA	**SA**	**1**

LES SYSTEMES DE NUMERATION ET CODES

I. Principes

Différents systèmes de numération sont utilisés en automatique. Les plus courants sont le système décimal (utilisé par l 'Homme), le système binaire (utilisé par la machine pour le traitement de l 'information) et le système hexadécimal (système intermédiaire et condensé).
Un système est défini par sa base.

Les bases

Une base B caractérise un système de numération dans lequel tout nombre N peut s 'écrire :

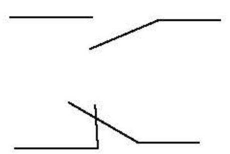
$$N_{(B)} = m_nB^n + m_{n-1}B^{n-1} + ... + m_1B + M_0B$$

- B : la base,
- m : valeur du coefficient (suivant la base),
- n : le rang de chaque chiffre (0,1,2,3...n-1,n).

Le système binaire

- Base : 2
- Valeur du coefficient m : 0 , 1

Exemple : signal reçu



0

1

Application :

1. Définir un nombre décimal à partir d 'une valeur binaire

Soit le nombre **1001110011**$_{(2)}$
Donner la valeur du nombre en base décimale(10)

On applique la méthode des poids,

ce qui donne :

1001110011$_{(2)}$ =

1	0	0	1	1	1	0	0	1	1
									2^0
									1

LES SYSTEMES DE NUMERATION ET CODES

Le système binaire (suite)

2. Définir un nombre binaire à partir d 'une valeur décimale

- Soit le nombre $325_{(10)}$, Donner sa valeur binaire

On divise le nombre par la base désirée, ici la base 2 (binaire)

$$\begin{array}{r|l} 325 & 2 \\ \hline & \end{array}$$

Réponse

calcul sur les nombres binaires

addition

0+0 =0
0+1 = 1
1+0 =1
1+1 =10

exemple :

110 + 11 = ?

$$\begin{array}{r} 1\,1\,0 \\ +\ \ 1\,1 \\ \hline = \end{array}$$

Le système décimal

- Base : 10
- Valeur du coefficient m : 0 , 1, 2, 3, 4, 5, 6, 7, 8, 9

$$N_{(B)} = m_n B^n + m_{n-1} B^{n-1} + \ldots\ldots + m_1 B + M_0 B$$

Exemple : :

Le nombre décimal $481_{(10)} = 4 .10^2 + 8 . 10^1 + 1\ 10^0$

Le nombre de combinaisons est égal à Bd, avec B = base, d= nombre de digits.

Pour 4 chiffres décimaux, le nombre de combinaisons donne 10^4 = 10 000 donc la possibilité de compter de 0 à 9999.

LES SYSTEMES DE NUMERATION ET CODES

Le MOT

- Le MOT est un ensemble de Bits (variables binaires), le format d 'un MOT est le plus souvent de 16 bits, quelquefois de 32 bits appelé Mot Double (MD).

Exemple : Mot de 16 bits

Octet de poids fort								Octet de poids faible								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Rang
1	1	0	0	0	0	1	1	0	1	1	0	1	0	0	1	MOT

Le système Hexadécimal

- Base : 16
- Valeur du coefficient m : 0 , 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

$$N_{(B)} = m_n B^n + m_{n-1} B^{n-1} + \ldots\ldots + m_1 B + M_0 B$$

Exemple : :

Le nombre décimal $A3F_{(16)} = A\,.16^2 + 3\,.\,16^1 + F\,16^0$
Pour un mot de 3 variables, il existe $16^3 = 4096$ combinaisons.

Décimal	Binaire pur	Héxadécimal
00	00000	00
01	00001	01
02	00010	02
03	00011	03
04	00100	04
05	00101	05
06	00110	06
07	00111	07
08	01000	08

Décimal	Binaire pur	Héxadécimal
09	01001	09
10	01010	0A
11	01011	0B
12	01100	0C
13	01101	0D
14	01110	0E
15	01111	0F
16	10000	10

LES SYSTEMES DE NUMERATION ET CODES

Les codes

Le code GRAY ou binaire réfléchi

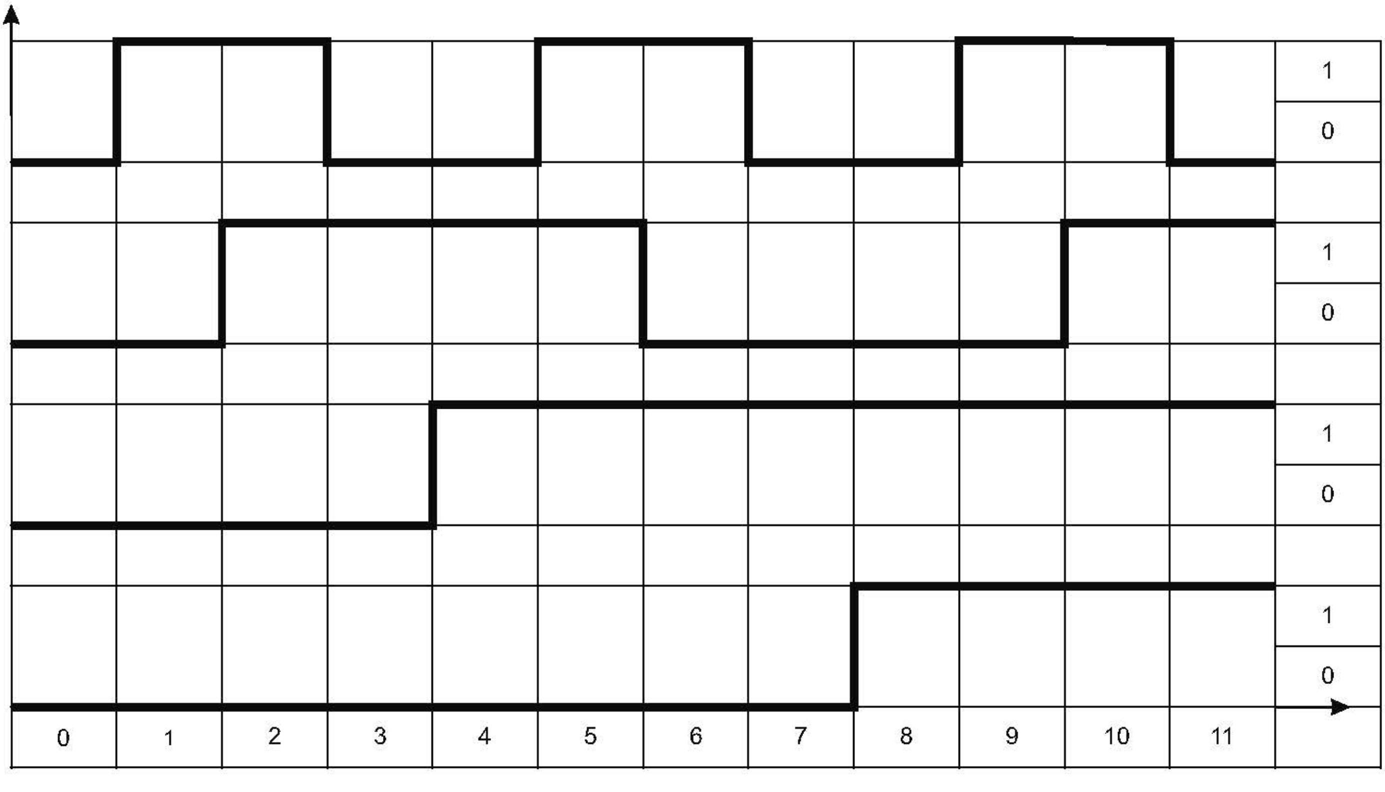
Le code Gray n 'autorise le changement d état que d 'une variable dans une combinaison
Un changement d 'état de variables est source d 'aléas de fonctionnement lorsqu 'il est utilisé dans la représentation de grandeurs physiques (codage de position : règle optique, codeur...).

Exemple d 'aléas :
Soit la position initiale = $3_{(10)}$ - valeur qui peut être 3 mm,

Sa valeur binaire est 0011.

La position suivante = $4_{(10)}$, en binaire sa valeur est 0100.

On observe que 2 valeurs binaires ont changé, pour résoudre ce problème, on utilise le code GRAY.



Le code GRAY est un code non pondéré, il n 'est pas utilisé pour des calculs arithmétiques.

LES SYSTEMES DE NUMERATION ET CODES

Application

Compléter la table de vérité suivante :

a	b	c	S

Lycée privé Théophile Legrand	BTS CRSA	**SA**	**6**

LES SYSTEMES DE NUMERATION ET CODES

Application

Réaliser les changements de bases pour les valeurs suivantes :

$192_{(10)}$ =…………………… $_{(2)}$

$34_{(8)}$ =……………………… $_{(10)}$

$49_{(10)}$ =……………………… $_{(2)}$

=……………………… $_{(8)}$

1011110000110100 = ……………………… $_{(10)}$

=……………………… $_{(16)}$

Diagrammes Logiques

OU

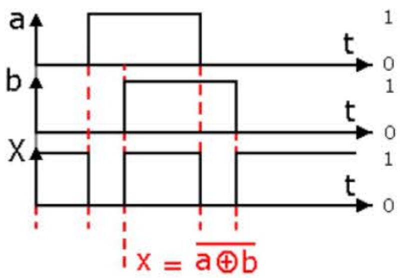
NOR- « OU NON »

ET

NAND – « ET NON »

XOR – « OU exclusif »

XNOR – « Identité »



Elle sera utilisé dans les circuits arithmétiques ou en automatisme pour tester l'égalité de deux variables

Lycée privé Théophile Legrand	BTS CRSA	**SA**	**7**
LES SYSTEMES DE NUMERATION ET CODES			

<u>Le code BCD (Binaire Codé Décimal)</u>

- Les systèmes digitaux utilisent tous les nombres binaires pour leurs opérations internes.

- La numération décimale étant la plus utilisée, cela suppose une conversion entre numération en décimale et la numération binaire.

- Le code BCD utilise 4 bits pour définir une valeur décimale

ex : $2_{(10)} = 0010_{(BCD)}$

- Un digit décimal peut atteindre la valeur (9), il faut donc nécessairement 4 bits pour coder chaque digit

Applications

- Coder les valeurs décimales suivante en codage BCD

$874_{(10)}$ = ----------------------------- $_{(BCD)}$

$932_{(10)}$ = ---------------------------------- $_{(BCD)}$

$1317_{(10)}$ = --- $_{(BCD)}$

- Coder les valeurs décimales suivante en codage binaire

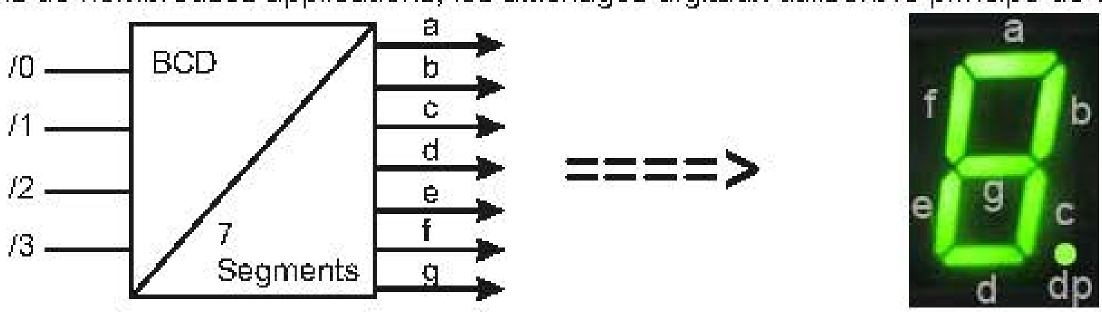
$874_{(10)}$ = ----------------------------- $_{(2)}$

$932_{(10)}$ = ---------------------------------- $_{(2)}$

$1317_{(10)}$ = --- $_{(2)}$

<u>Transcodage BCD ===> 7 segments</u>

Dans de nombreuses applications, les affichages digitaux utilisent le principe de codage BCD.



LES SYSTEMES DE NUMERATION ET CODES

Compléter le tableau suivant :

Variables d 'entrées				Affichage	Variables de sorties						
/3	/2	/1	/0	segments	a	b	c	d	e	f	g
0	0	0	0								
0	0	0	1								

<u>Transcodage binaire pur ===> binaire réfléchi</u>

<u>Rappel</u> : Le OU exclusif (principe du va et vient)

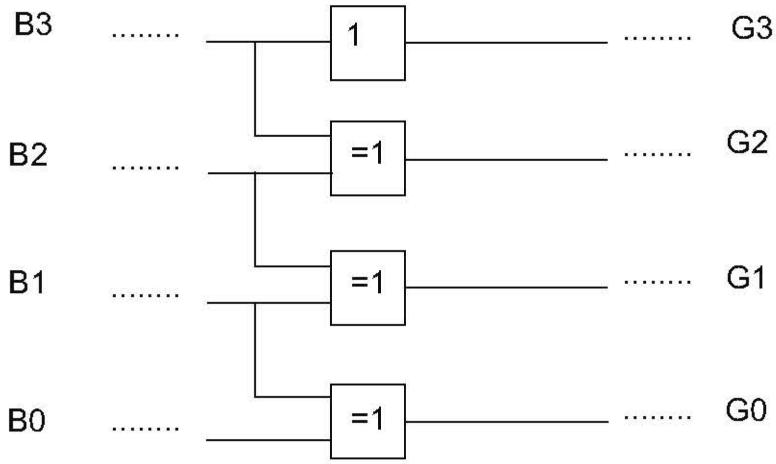
équation : $S = \bar{a}b + a\bar{b}$

$= a \oplus b$

a	b	S
0	0	
0	1	
1	1	
1	0	

Le logigramme suivant permet de transcoder un nombre binaire pur à 4 bits en nombre binaire réfléchi (GRAY).

Réaliser le codage de la valeur **5** en code binaire et en code gray



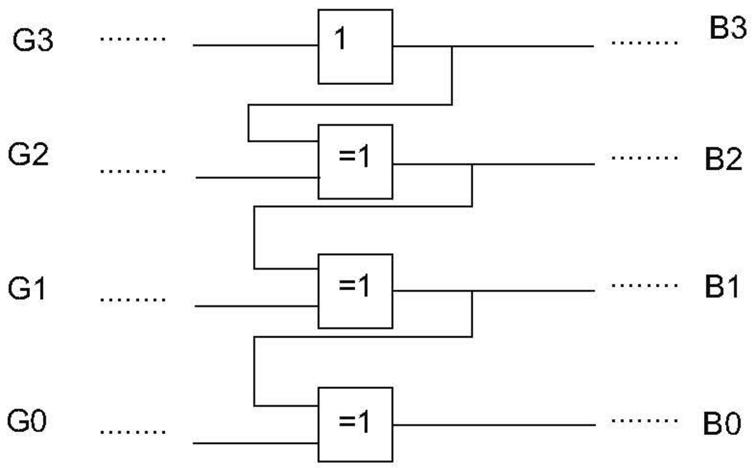
Ce type de transcodage est utilisé pour le décodage des informations fournies par les capteurs rotatifs (codeurs absolus).

LES SYSTEMES DE NUMERATION ET CODES

Transcodage binaire réfléchi ===> binaire pur

Le logigramme suivant permet de transcoder un nombre binaire réfléchi (GRAY) à 4 bits en nombre binaire pur.

Réaliser le codage de la valeur 6 en code binaire et en code gray



LES SYSTEMES DE NUMERATION ET CODES

APPLICATIONS

Transcodage binaire pur ===> binaire réfléchi

$(12)_{10}$ = ($)_2$ --------------------->($)_G$

$(27)_{10}$ = ($)_2$ --------------------->($)_G$

$(51)_{10}$ = ($)_2$ --------------------->($)_G$

rappel : $G = (n \oplus 2n) / 2$

LES SYSTEMES DE NUMERATION ET CODES

Les nombres signés

a. Complément d 'un nombre binaire

- Il représente l 'opposé d 'un nombre binaire, il est noté $\overline{X}$.

ex : 1000111001 a pour complément ....

b. Complément à 2 d 'un nombre binaire

- Le complément à 2 d 'un nombre binaire signé (noté sur x bits), permet de déterminer la valeur négative.
- Il est aussi appelé : **complément vrai**.

Complément à 2 = complément à 1 + 1

Principe :

- Afin de réaliser le calcul de l 'opposé d 'un nombre entier, il faut déterminer la **longueur de la représentation** (ex : 8, 16, 32 bits....) et la valeur absolue | X | du nombre à coder.

Formule :

$$2^n - |X| = -X$$

- écriture du nombre (-1) sur 8 bits

----> $2^8 - |-1| = 256 - 1 = 255$ ------------> $11111111_{(2)} = -1$

- Déterminer la valeur binaire de (- 629) sur 16 bits , utiliser les 2 méthodes.

LES SYSTEMES DE NUMERATION ET CODES

Les nombres signés

Soit l 'octet suivant : 0 0 0 0 0 0 0 0

—

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

Le 4 juin 1996, une fusée Ariane 5 a explosé 40 secondes après l'allumage. La fusée et son chargement avaient coûté 500 millions de dollars. La commission d'enquête a rendu son rapport au bout de deux semaines. Il s'agissait d'une erreur de programmation dans le système inertiel de référence. À un moment donné, un nombre codé en virgule flottante sur 64 bits (qui représentait la vitesse horizontale de la fusée par rapport à la plate-forme de tir) était converti en un entier sur 16 bits. Malheureusement, le nombre en question était plus grand que 32768 (le plus grand entier que l'on peut coder sur 16 bits) et la conversion a été incorrecte

LES SYSTEMES DE NUMERATION ET CODES

APPLICATION

—

Quel est dans un octet le bit qui sert de bit de signe ?
A. Bit 0
B. Bit 1
C. Bit 7
D. Bit 8
E. Bit 15

Quel est la plus grande valeur positive que l 'on puisse écrire en nombre signé sur 8 bits ?
A. 99
B. 127
C. 128
D. 255
E. 256

Quelle est la valeur décimale du nombre signé FF(16) ? Soit 1111 1111 en binaire ?

Que vaut le nombre NON SIGNE FF(16) ? Soit 1111 1111 en binaire ?

Que vaut le nombre signé 80(16) ou 1000 0000 en base 2 ?

Que donne le calcul 0 moins 1 s'il est fait en hexadécimal avec des nombres de 1 octet ?

Que donne le calcul 0 moins 1, s'il est fait en hexadécimal avec des nombres de 2 octets ?

Le code de 8 bits 7F en hexadécimal a-t-il la même valeur si ce nombre est signé ou non ?

Quel est la plus grande valeur positive que l'on puisse écrire avec un nombre signé de 2 octets ?

LES SYSTEMES DE NUMERATION ET CODES

Représentation des nombres réels

Conversion binaire- décimal à virgule fixe

—

En base 10, le nombre 547,568 peut être interpréter par le calcul suivant :

$$5.10^{2} + 4.10^{1} + 7.10^{0} + 5.10^{-1} + 6.10^{-2} + 8.10^{-3}$$

En base 2, il en est de même, l'expression 101 , 1101 peut s 'écrire :

$$1.2^{2} + 0.2^{1} + 1.2^{0} + 1.2^{-1} + 1.2^{-2} + 0.2^{-3} + 1.2^{-4}$$
$$=$$
$$4 + 1 + 0,5 + 0,125 + 0,0625$$
$$=$$
$$5,6875$$

Soit le nombre binaire suivant : 1011 , 101

On désire déterminer sa valeur en décimale, on utilise la formule suivante :

$$N , m = \text{bit } n x 2^{n} .... + \text{bit } x 2^{3} + \text{bit } x\ 2^{2} + \text{bit } x\ 2^{1} + \text{bit } x\ 2^{0} + \text{bit } x\ 2^{-1} + \text{bit } x\ 2^{-2} + ... \text{ bit } x\ 2^{-n}$$

1011 , 101 = ...

Conversion décimal - binaire virgule fixe

Le passage de base 10 en base 2 est plus subtil. Par exemple : convertissons 1234,347 en base 2.

La partie entière se transforme : 1234 = 10011010010

On transforme la partie décimale selon le schéma suivant :

$0,347 \cdot 2 = 0,694$	0,347 = 0,0...
$0,694 \cdot 2 = 1,388$	0,347 = 0,01...
$0,388 \cdot 2 = 0,766$	0,347 = 0,010...
$0,766 \cdot 2 = 1,552$	0,347 = 0,0101..
$0,552 \cdot 2 = 1,104$	0.347 = 0,01011...
$0,104 \cdot 2 = 0,208$	0,347 = 0,010110...
$0,208 \cdot 2 = 0,416$	0,347 = 0,0101100...
$0,416 \cdot 2 = 0,832$	0,347 = 0,01011000...
$0,832 \cdot 2 = 1,664$	0,347 = 0,010110001...
$0,664 \cdot 2 = 1,328$	0,347 = 0,0101100011...
$0,328 \cdot 2 = 0,656$	0,347 = 0,01011000110............

On continue ainsi jusqu'à la précision désirée

LES SYSTEMES DE NUMERATION ET CODES

Le 25 février 1991, pendant la Guerre du Golfe, une batterie américaine de missiles Patriot, à Dharan (Arabie Saoudite), a échoué dans l'interception d'un missile Scud irakien.
Le Scud a frappé un baraquement de l'armée américaine et a tué 28 soldats.

La commission d'enquête a conclu à un calcul incorrect du temps de parcours, dû à un problème d'arrondi.
Les nombres étaient représentés en virgule fixe sur 24 bits. Le temps était compté par l'horloge interne du système en dixième de seconde.
Malheureusement, 1/10 n'a pas d'écriture finie dans le système binaire : 1/10 = 0,1 (dans le système décimal) = 0,0001100110011001100110011... (dans le système binaire).

L'ordinateur de bord arrondissait 1/10 à 24 chiffres, d'où une petite erreur dans le décompte du temps pour chaque 1/10 de seconde.
Au moment de l'attaque, la batterie de missile Patriot était allumée depuis environ 100 heures, ce qui a entraîné une accumulation des erreurs d'arrondi de 0,34 s. Pendant ce temps, un missile Scud parcourt environ 500 m, ce qui explique que le Patriot soit passé à côté de sa cible.

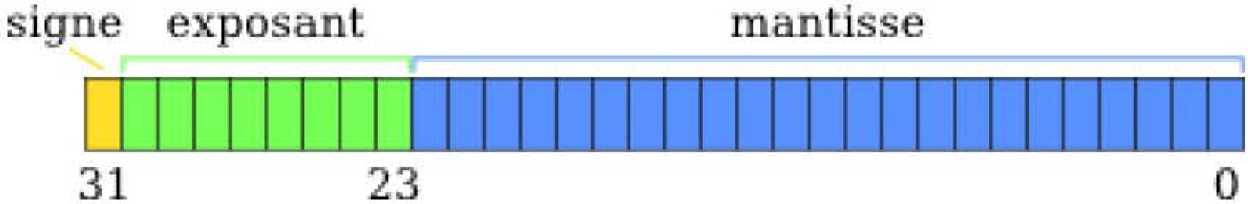
Application

Transformez 0,562510 en base 2.

Transformez 0,1510 en base 2.

La norme IEEE 754

La norme IEEE 754 définit la façon de coder un nombre réel. Cette norme se propose de coder le nombre sur 32 bits et définit trois composantes :
- le signe est représenté par un seul bit, le bit de poids fort,
- l'exposant est codé sur les 8 bits consécutifs au signe,
- la mantisse (les bits situés après la virgule) sur les 23 bits restants.



Certaines conditions sont toutefois à respecter pour les exposants :

- l'exposant 00000000 est interdit.

- l'exposant 11111111 est interdit. On s'en sert toutefois pour signaler des erreurs, on appelle alors cette configuration du nombre NaN, ce qui signifie « Not a number ».

- il faut rajouter 127 (01111111) à l'exposant pour une conversion de décimal vers un nombre réel binaire. Les exposants peuvent ainsi aller de −254 à 255.

- La formule d'expression des nombres réels est ainsi la suivante : (−1)^S * 2^(E − 127) * (1 + F)

- S est le bit de signe et l'on comprend alors pourquoi 0 est positif (−1^0=1),
- E est l'exposant auquel on doit bien ajouter 127 pour obtenir son équivalent codé,
- F est la partie fractionnaire.

Exemple

Traduisons en binaire, en utilisant la norme IEEE 754, le nombre −6,625.

- Codons d'abord la valeur absolue en binaire : $6{,}625_{10} = 110{,}1010_2$

- Nous mettons ce nombre sous la forme : 1, partie fractionnaire
$110{,}1010 = 1{,}101010 \cdot 2^2$ (2^2 décale la virgule de 2 chiffres vers la droite)

- La partie fractionnaire étendue sur 23 bits est donc 101 0100 0000 0000 0000 0000.

Exposant = 127 + 2 = 129_{10} = $1000\ 0001_2$

Signe	Exposant								Mantisse																						
1	1	0	0	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

En hexadécimal : C0 D4 00 00.

LES SYSTEMES DE NUMERATION ET CODES

<u>Programmation d 'un Octet d 'entrée sur API S7-1200</u>

Un Octet d 'entrée : % IB ... se programme sur 8 bits (un octet)

% IB 0 correspond aux entrées de 0 à 7 sur le premier module

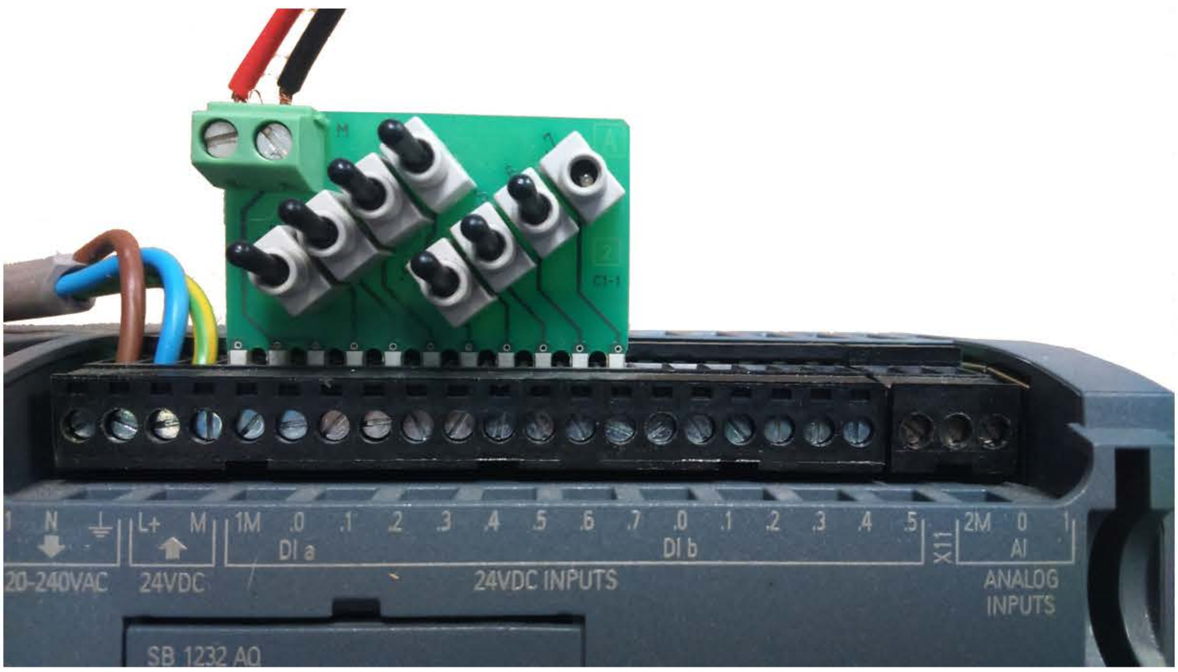
Cela permet de définir une valeur numérique sur un octet:

Application :

réaliser la conversion suivante $(255)_{10} = (...............)_8$

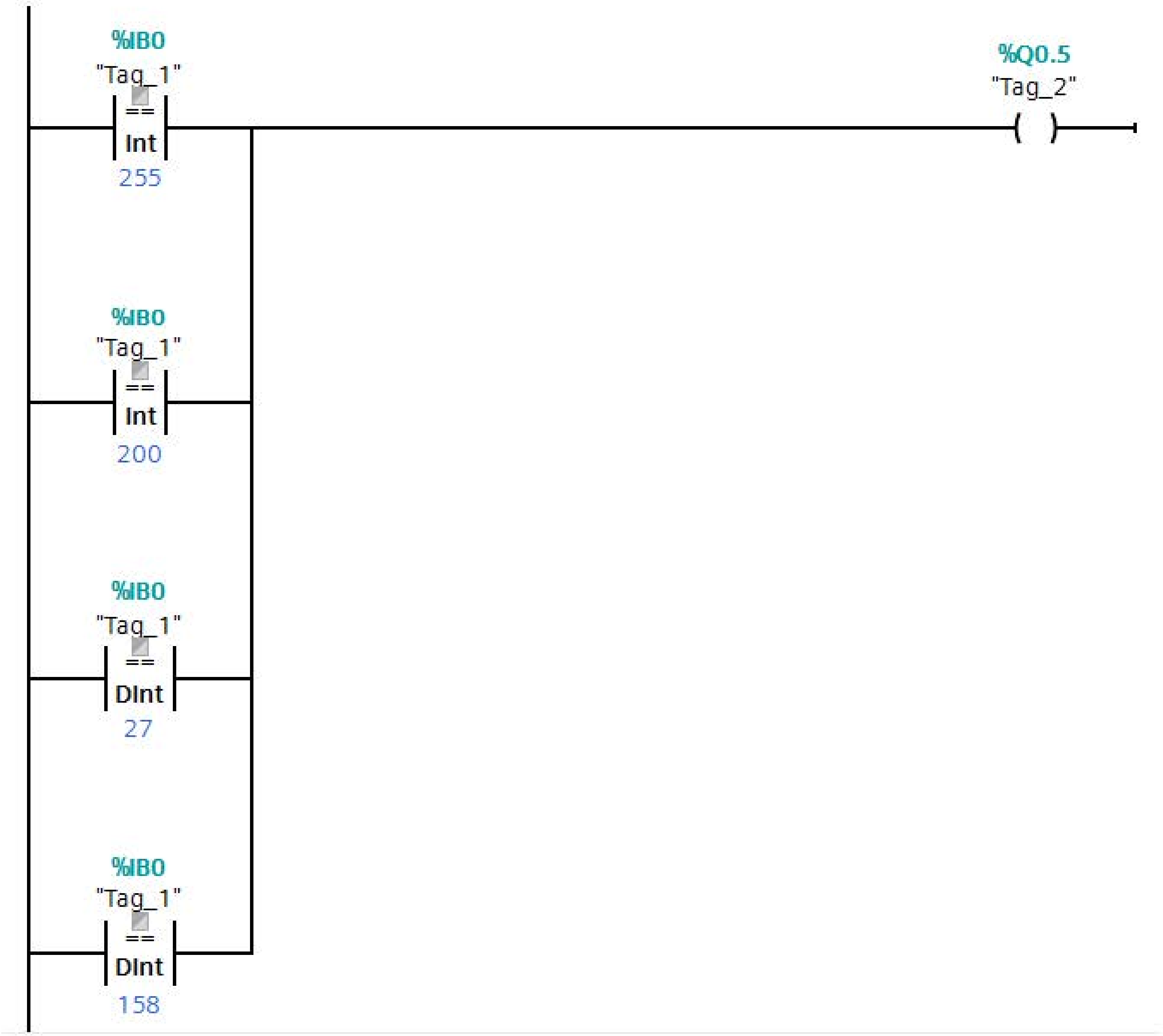
sur 8 bits cela donne en binaire $(.................................)_2$

A partir de ces informations, réaliser la ligne de programme suivante et indiquer quelles entrées activées pour valider la sortie



LES SYSTEMES DE NUMERATION ET CODES

Programmer cette ligne



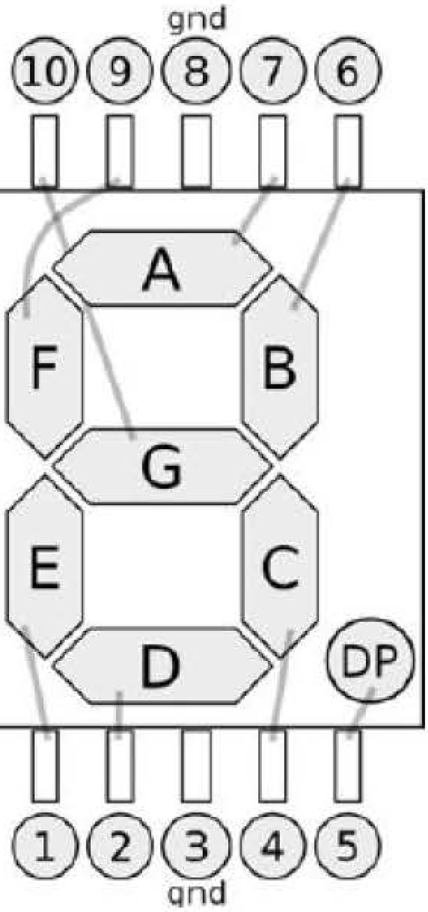
Définir les entrées à valider pour activer la sortie :

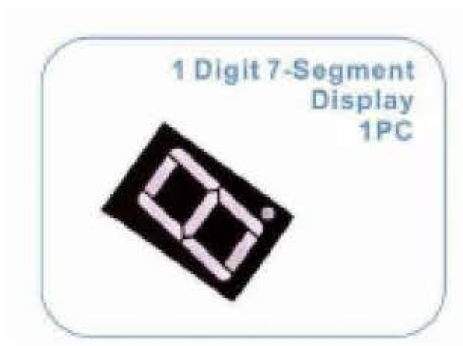
	I0.7	I0.6	I0.5	I0.4	I0.3	I0.2	I0.1	I0.0
255								
200								
27								
138								

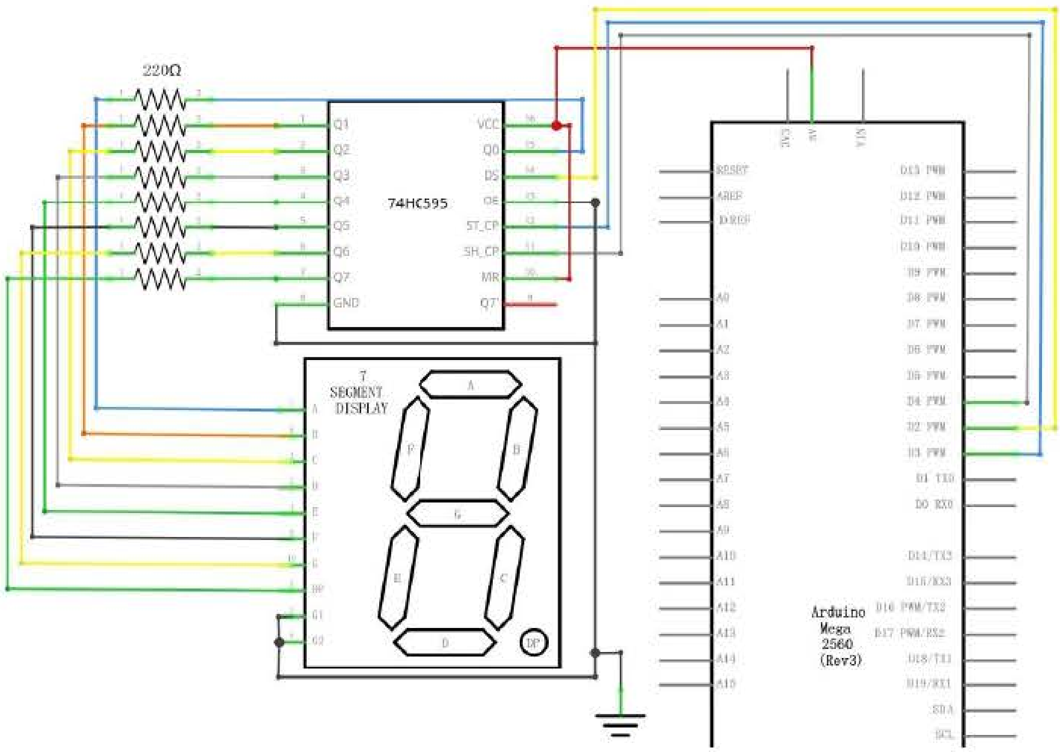
LES SYSTEMES DE NUMERATION ET CODES

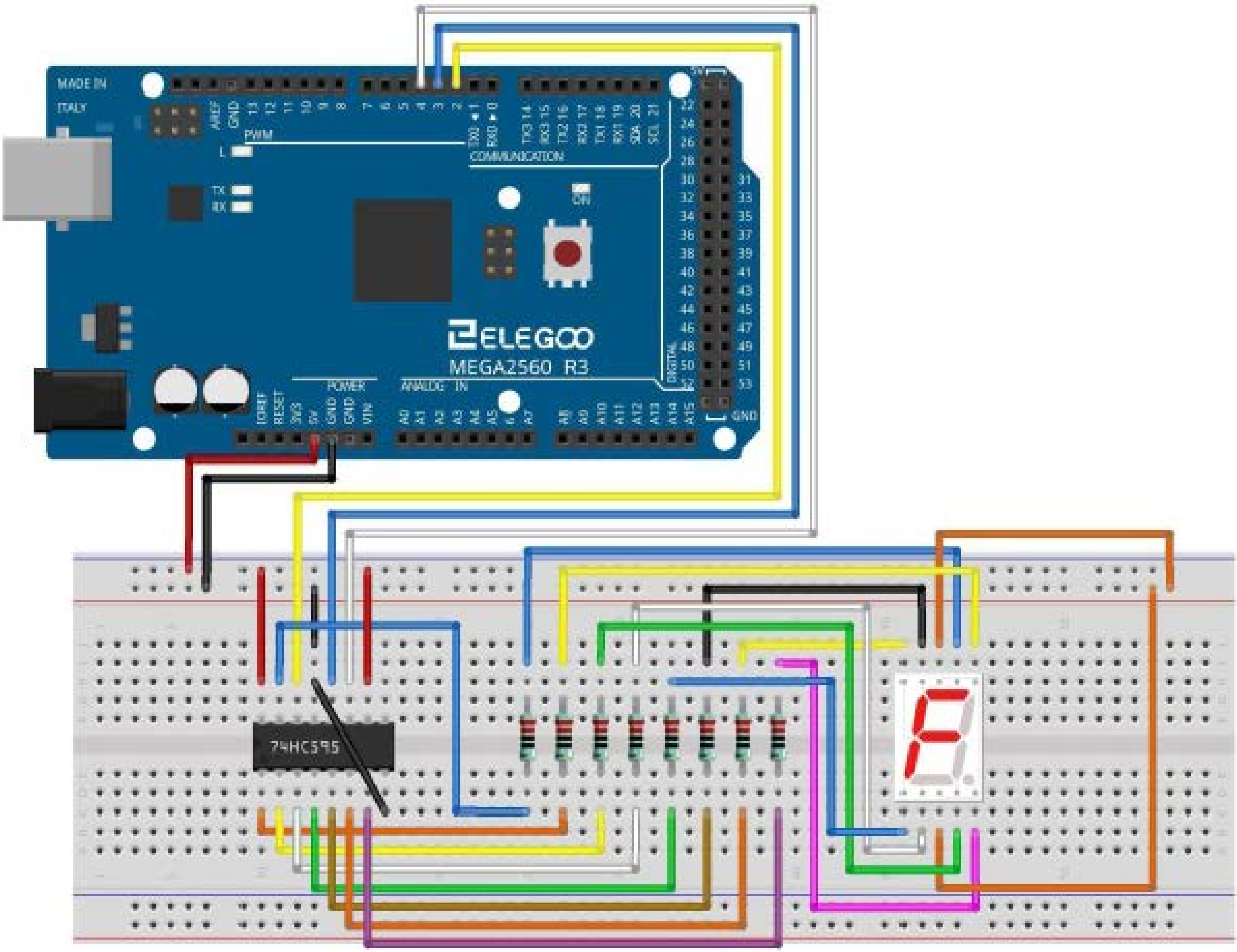
Présentation du composant Afficheur 7 segments

Diagramme des pins de l'afficheur:









ELEGOO
MEGA2560 R3
74HC595